

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління

(повна назва)

Кафедра Комп'ютерних інтелектуальних технологій та систем

(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти

перший (бакалаврський)

Мікросервісна система адаптивного керування мікрокліматом на основі
аналізу сенсорних даних

Виконав:

здобувач IV року навчання,
групи КІУКІ-21-10

Дмитро ДЕМЧЕНКО

(власне ім'я, прізвище)

Спеціальність 123 Комп'ютерна інженерія

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія

(повна назва освітньої програми)

Керівник проф. каф. КІТС Наталія АКСАК

(посада, власне ім'я, прізвище)

Допускається до захисту

Зав. кафедри

Олег РУДЕНКО

2025 р.

Харківський національний університет радіоелектроніки

Факультет	Комп'ютерної інженерії та управління
Кафедра	Комп'ютерних інтелектуальних технологій та систем
Рівень вищої освіти	перший (бакалаврський)
Спеціальність	123 Комп'ютерна інженерія
Тип програми	освітньо-професійна
Освітня програма	Комп'ютерна інженерія

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Демченко Дмитру Едуардовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Мікросервісна система адаптивного керування мікрокліматом на основі аналізу сенсорних даних

затверджена наказом по університету від “ 21 ” травня 2025 р. № 399СТ

2. Термін подання здобувачем роботи до екзаменаційної комісії 14.06.2025

3. Вхідні дані до роботи Дані з IoT-сенсорів мікроклімату

Історичні дані для навчання моделі

Конфігураційні параметри системи

Розробка основних мікросервісів:

Інтеграція сенсорних даних з IoT-пристроїв через MQTT-протокол.

Застосування LSTM-моделі для прогнозування параметрів мікроклімату

4. Перелік питань, що потрібно опрацювати в роботі _____

Аналіз предметної області

Огляд існуючих архітектурних рішень

Формалізація вимог до системи

Проектування архітектури системи на основі мікросервісної моделі.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) 10 слайдів

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк виконання / терміни етапів роботи	Примітка
1	Огляд і аналіз сучасного стану розглянутої проблеми, а також існуючих методів і засобів вирішення задач кваліфікаційної роботи	26.05.2025 – 31.05.2025	Виконано
2	Проектування архітектури додатку	01.06.2025 – 03.06.2025	Виконано
3	Створення додатку	04.06.2025 – 07.06.2025	Виконано
4	Аналіз та налагодження роботи додатку	08.06.2025 – 09.06.2025	Виконано
5	Оформлення матеріалів кваліфікаційної роботи	10.06.2025 – 14.06.2025	Виконано

Дата видачі завдання 26.05.2025

Здобувач _____
(підпис)

Керівник роботи _____
(підпис)

Проф. Наталія АКСАК
(посада, ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 62 с., 11 рис., 1 табл., 1 дод., 21 джерел.

МІКРОКЛІМАТ, МІКРОСЕРВІСНА АРХІТЕКТУРА, СЕНСОРИ, LSTM, ПРОГНОЗУВАННЯ, MQTT, КЕРУВАННЯ, DOCKER, АДАПТИВНІ СИСТЕМИ, ІОТ

У роботі розглянуто проєктування та реалізацію мікросервісної системи адаптивного керування мікрокліматом на основі аналізу сенсорних даних. Розроблена система призначена для підтримання оптимальних умов середовища (температури, вологості) у закритих приміщеннях, таких як теплиці або розумні будівлі. Основу системи становлять незалежні мікросервіси, що відповідають за збір даних, збереження, аналіз, прогнозування та керування виконавчими пристроями. Для обробки часового ряду сенсорних даних та прогнозування кліматичних параметрів використано LSTM-модель глибокого навчання, що дозволяє формувати проактивні рішення для стабілізації мікроклімату. Реалізація здійснена на базі контейнеризованих сервісів з використанням Docker, MQTT, Prometheus та Grafana. Проведено кількісний аналіз точності моделі на основі метрик MAE, MSE та R^2 , а також побудовано графіки порівняння фактичних і прогнозованих значень. Результати показали високу точність прогнозів та стабільність роботи системи. Робота містить пояснювальну записку, UML-діаграми, схеми даних та графіки оцінки якості.

ABSTRACT

Explanatory note of qualification work 62 pages, 11 figures, 1 tables, 1 appendices, 21 sources.

MICROCLIMATE, MICROSERVICE ARCHITECTURE, SENSORS, LSTM, FORECASTING, MQTT, CONTROL, DOCKER, ADAPTIVE SYSTEMS, IOT

The bachelor's thesis presents the design and implementation of a microservice-based adaptive microclimate control system based on sensor data analysis. The system aims to maintain optimal environmental conditions (temperature, humidity) in enclosed spaces such as greenhouses and smart buildings. The architecture consists of independent microservices responsible for data ingestion, storage, analytics, forecasting, and actuator control. To process time-series sensor data and forecast environmental parameters, an LSTM deep learning model is used. This model enables proactive control decisions for microclimate stabilization. The system is implemented using Docker-based containers, MQTT for sensor communication, Prometheus and Grafana for monitoring. A quantitative accuracy evaluation is carried out using MAE, MSE, and R^2 metrics, supported by comparative graphs between actual and predicted values. The results confirm high forecast accuracy and robust system performance. The project includes a technical report, UML diagrams, data flow schemes, and evaluation plots.

ЗМІСТ

Вступ	8
1 Актуальність задачі та аналіз предметної області	12
1.1 Аналіз предметної області	12
1.2 Актуальність обраної теми	13
1.3 Огляд існуючих рішень	14
1.4 Визначення проблем та методів їх рішення	21
1.5 Постановка задачі	22
2 Мікросервісна архітектура та сіті-фермерство	24
2.1 Основні характеристики мікросервісів	24
2.2 Сіті-фермерство	25
3 Проектування мікросервісної системи адаптивного керування мікрокліматом	31
3.1 План розробки мікросервісної системи	32
3.3 Архітектура системи	35
3.4 Проектування мікросервісів	37
4 Проектування модулів аналізу даних	42
4.1 Основні функції аналітики	42
4.2 LSTM-модель для прогнозування параметрів мікроклімату	44
Висновки	54
Перелік використаних джерел	55
Додаток А Графічний матеріал кваліфікаційної роботи	П

омилка! Закладку не визначено.

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

API — Application Programming Interface (інтерфейс прикладного програмування)

CPU — Central Processing Unit (центральний процесор)

Docker — система контейнеризації застосунків

IoT — Internet of Things (Інтернет речей)

LSTM — Long Short-Term Memory (довготривала короткочасна пам'ять)

MAE — Mean Absolute Error (середня абсолютна похибка)

MSE — Mean Squared Error (середньоквадратична похибка)

MQTT — Message Queuing Telemetry Transport (протокол телеметрії)

R^2 — коефіцієнт детермінації

REST — Representational State Transfer (архітектурний стиль вебсервісів)

RNN — Recurrent Neural Network (рекурентна нейронна мережа)

UML — Unified Modeling Language (уніфікована мова моделювання)

ВСТУП

Забезпечення оптимальних параметрів мікроклімату в приміщеннях є ключовим чинником для досягнення комфортних умов перебування людей, підвищення продуктивності праці, а також ефективного функціонування чутливого обладнання. У традиційних системах керування мікрокліматом домінують централізовані підходи, що часто характеризуються недостатньою гнучкістю, обмеженою масштабованістю та низькою здатністю до адаптації у динамічному середовищі. Це зумовлює потребу у впровадженні новітніх архітектурних і технологічних рішень.

Застосування мікросервісної архітектури у поєднанні з аналізом сенсорних даних та методами машинного навчання дозволяє створити розподілені інтелектуальні системи, здатні до самостійного ухвалення рішень на основі поточних показників довкілля. Такі системи підтримують адаптивне регулювання кліматичних параметрів в режимі реального часу, що забезпечує підвищену енергоефективність, надійність та модульність.

Адаптивне керування мікрокліматом – це підхід до автоматизації підтримання оптимальних умов середовища (температури, вологості, тощо) у приміщеннях або теплицях на основі даних від датчиків. Такий контроль важливий для енергоефективності «розумних» будівель та підвищення врожайності в сільському господарстві. Сучасні системи все частіше використовують архітектуру мікросервісів та Інтернет речей (IoT) для модульності та масштабованості, а також алгоритми машинного навчання (МН) для прогнозування та адаптивного прийняття рішень.

Для підготовки та реалізації цієї роботи було використано широкий спектр інформаційних джерел:

Офіційні документи: технічні стандарти, специфікації мікросервісної архітектури, протоколи зв'язку IoT (MQTT, CoAP), офіційна документація Docker, ASP.NET Core, Prometheus та інші.

Наукові джерела: публікації у рецензованих журналах та конференціях (IEEE, MDPI, Elsevier, Springer), що висвітлюють теоретичні та прикладні аспекти адаптивного управління мікрокліматом, обробки сенсорних даних та використання ML-алгоритмів.

Бібліографічні огляди: роботи, присвячені аналізу архітектурних шаблонів у розумному землеробстві, smart building, цифровій трансформації агросектору та індустрії 4.0.

Прикладна література та онлайн-ресурси: технічні блоги, white papers, кейс-стаді розробників індустріальних платформ (FIWARE, BEMS, IRRISENS), GitHub-репозиторії та інструкції до впровадження мікросервісних екосистем.

Такий комплексний підхід до вибору джерел дозволив поєднати теоретичну обґрунтованість, практичну реалізованість та актуальність сучасних технологій у галузі адаптивного керування середовищем.

Методологія дослідження базується на міждисциплінарному підході, який поєднує:

Системний аналіз предметної області мікрокліматичного контролю: визначення ключових змінних, обмежень і вимог до системи.

Методи програмної інженерії: зокрема принципи побудови мікросервісної архітектури, застосування шаблонів REST API, організація взаємодії через брокери повідомлень, реалізація логування, моніторингу та контейнеризації (Docker).

Інструментарій обробки даних: попередня обробка сенсорних потоків (фільтрація, агрегація, злиття), побудова моделей на основі історичних даних, використання бібліотек ML (Scikit-learn, TensorFlow, PyTorch).

Моделювання і прототипування: створення та тестування прототипу у симуляційному середовищі з можливістю подальшого розгортання в Docker.

Емпіричне тестування: перевірка працездатності системи шляхом генерації тестових сценаріїв і збору метрик ефективності (затримка, стабільність параметрів, кількість рішень, споживання ресурсів).

Поєднання інженерних та наукових методів дало змогу створити не лише працездатну систему, а й узагальнити підходи до адаптивного керування на основі мікросервісної логіки.

1 АКТУАЛЬНІСТЬ ЗАДАЧІ ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної області

Управління мікрокліматом у замкнених просторах є критично важливим завданням для багатьох галузей, включаючи аграрний сектор (вертикальні ферми, теплиці), медичні установи, складські приміщення, офісні будівлі та освітні заклади. Основна мета таких систем — забезпечення стабільних та оптимальних умов навколишнього середовища для підтримання життєдіяльності, збереження продукції або ефективного функціонування технологічних процесів. Ключові параметри мікроклімату включають температуру, вологість, рівень CO₂, освітленість і вентиляцію.

Традиційні системи автоматизації, як правило, базуються на централізованому управлінні та слабо адаптуються до змін у зовнішньому або внутрішньому середовищі. Вони мають обмежену гнучкість, низьку масштабованість і не забезпечують достатню точність регулювання в умовах високої динаміки даних. Водночас поява Інтернету речей (IoT), edge-обчислень, штучного інтелекту та хмарних технологій створила нові можливості для побудови розподілених інтелектуальних систем.

Мікросервісна архітектура є одним із найперспективніших підходів до розробки таких систем. Вона дозволяє поділити функціональність на незалежні сервіси, кожен з яких виконує окреме завдання — збір сенсорних даних, аналіз інформації, прийняття рішень, управління виконавчими пристроями тощо. Це забезпечує гнучкість масштабування, швидке оновлення компонентів, адаптивність до різних умов експлуатації та високу надійність.

Окремо слід виділити потенціал інтеграції методів машинного навчання — для прогнозування змін мікроклімату, виявлення аномалій, формування адаптивних стратегій управління та енергетичної оптимізації. В контексті розумного міського

середовища, «зелених» технологій та індустрії 4.0, мікросервісні системи керування мікрокліматом виступають ключовими інструментами цифрової трансформації.

1.2 Актуальність обраної теми

У контексті стрімкого розвитку технологій Інтернету речей (IoT) та вимог до сталого енергоспоживання, особливої актуальності набувають системи, які не лише автоматизують керування мікрокліматом, а й оптимізують його з урахуванням змінних умов середовища. Розробка адаптивної системи, що використовує аналітичні модулі для обробки сенсорної інформації та прийняття рішень, є важливим кроком до створення "розумних" середовищ — як у побутовому, так і в промисловому сегменті.

Актуальність використання мікросервісної системи адаптивного керування мікрокліматом полягає у вирішенні низки сучасних проблем у сфері енергоефективності, автоматизації та сталого розвитку середовищ, зокрема в аграрному секторі, «розумних» будівлях і замкнених просторах. Основні аргументи щодо актуальності:

Гнучкість та масштабованість. Мікросервісна архітектура дозволяє незалежно розгортати, оновлювати та масштабувати окремі сервіси (наприклад, контроль температури, вологості, CO₂, освітлення) відповідно до конкретних потреб об'єкта чи зміни кліматичних умов.

Інтеграція з IoT та хмарними технологіями. Система легко інтегрується з мережею сенсорів та edge/cloud-обчисленнями, що дозволяє оперативно аналізувати дані в реальному часі та приймати адаптивні рішення на основі штучного інтелекту або логічних правил.

Підвищення енергоефективності. Завдяки точному контролю та адаптації до динамічних умов середовища, система забезпечує економне споживання енергії, що особливо актуально у контексті енергетичної кризи та цілей сталого розвитку.

Забезпечення стабільних умов для технологічних процесів. У таких сферах, як рослинництво у закритому ґрунті (city farms, теплиці), зберігання продукції,

лабораторії чи лікарні, контроль мікроклімату має критичне значення. Мікросервісна система дає змогу динамічно підтримувати необхідні параметри з високою точністю.

Модульність та повторне використання. Сервіси можна легко адаптувати для різних типів приміщень і застосувань без необхідності повного перепроєктування системи — що знижує витрати на розробку та обслуговування.

Підтримка аналітики та прогнозування. Завдяки накопиченню історичних даних і використанню машинного навчання, система здатна передбачати зміни умов і приймати проактивні рішення, що підвищує її інтелектуальні можливості.

1.3 Огляд існуючих рішень

Мікросервісна архітектура передбачає розподіл функціоналу системи на набір дрібних незалежних сервісів, що взаємодіють через чіткі інтерфейси. Для систем керування мікрокліматом це означає, що окремі сервіси можуть відповідати за збір даних з датчиків, обробку/аналіз даних, прийняття рішень щодо керування обладнанням (нагрівачами, вентиляторами тощо) та взаємодію з користувачем. Такий підхід забезпечує низку переваг:

Модульність і гнучкість. Нові компоненти легко додаються як окремі мікросервіси, а існуючі – замінюються або оновлюються без впливу на всю систему [1]. Наприклад, в роботі Naque et al. (2020) для системи керування кліматом будівлі кожна функція (збір даних, аналітика, інтерфейс) реалізована окремим сервісом, що спрощує розробку та масштабування [2].

Масштабованість. Мікросервіси можна розгортати незалежно один від одного в хмарі чи на локальних вузлах, масштабуючи саме ті компоненти, які потребують більше ресурсів. Для прикладу, переписана на мікросервіси платформа BEMS (Building Energy Management System) змогла одночасно контролювати до 3200 пристроїв проти 400 у моноліті завдяки розподіленню навантаження [2].

Стійкість до відмов. Збої окремого сервісу не призводять до падіння всієї системи. Як зазначається, якщо мікросервіс, що логує дані, виходить з ладу, інші

(наприклад, сервіс керування пристроями) продовжать працювати [2]. Це запобігає каскадним відмовам і підвищує надійність критичних функцій.

У дослідженнях останніх років пропонуються мікросервісні платформи для IoT у агротехніці та будівлях. Зокрема, платформа IRRISENS (2020) – IoT-система для зрошення у сільському господарстві – реалізована повністю на мікросервісах [3]. Кожен фермерський об'єкт має свій ізольований набір сервісів, що збирають дані ґрунту, врожаю та погоди, виконують прогнозування (на основі моделей евапотранспірації) і надсилають команди на зрошення[3].

У сфері розумних мікрокліматичних систем, які орієнтовані на замкнене середовище (теплиці, ферми, лабораторії), робота [12] заслуговує на особливу увагу. У ній запропоновано інтелектуальне керування мікрокліматом city-ферми на основі алгоритму Q-learning. Система функціонує за принципом підсиленого навчання: агент спостерігає за температурою і вологістю в камері вирощування, отримує винагороди за досягнення бажаних параметрів і вчиться вибирати найкращі дії для керування системою опалення/зволоження. На відміну від класичних регуляторів, Q-learning дозволяє адаптуватися до зміни зовнішніх умов і внутрішнього стану середовища без необхідності формулювання аналітичної моделі об'єкта керування. Експериментальні результати демонструють стабільну підтримку мікроклімату та високу ефективність керування в умовах невизначеності [12].

Ще одним прикладом реалізації є публікація [13], де розглянуто архітектуру системи контролю мікроклімату у замкнутому приміщенні. У роботі представлено модульну структуру системи з окремими підсистемами збору сенсорних даних, прийняття рішень, управління виконавчими пристроями та моніторингу. Особливу увагу приділено побудові UML-діаграм, опису потоків даних, а також забезпеченню гнучкості системи для подальшого масштабування. Пропонована архітектура дозволяє інтегрувати алгоритми штучного інтелекту, включно з Q-learning, в існуючу інфраструктуру.

Такі приклади демонструють тренд до впровадження інтелектуальних агентів і методів машинного навчання в архітектуру мікрокліматичних систем. Вони підтверджують практичну доцільність переходу від традиційних систем

регулювання до адаптивних самонавчальних моделей, які здатні забезпечувати ефективну та стійку підтримку параметрів середовища.

Архітектура IRRISENS інтегрується з хмарними сервісами (наприклад, модулем ORION Context Broker в рамках платформи FIWARE) для оркестрації мікросервісів та забезпечення інтеоперабельності [3]. Рішення протестовано на чотирьох комерційних фермах і показало ефективність в плануванні зрошення та конфіденційність даних кожного користувача через розділення сервісів [3].

У сфері розумних будівель мікросервіси також довели свою ефективність. Naque et al. запропонували хмарну SaaS-платформу BEMS (2020) у вигляді набору мікросервісів: окремо для інтерфейсу користувача, аналітики даних, планування розкладів пристроїв тощо[2]. Аналітичний сервіс виконує прогнозування енергоспоживання будівлі за допомогою моделей машинного навчання (наприклад, прогноз навантаження за історичними даними) [2]. Сервіс додатків може реалізовувати стратегії керування, такі як demand response – зменшення споживання у часи пікового навантаження, шляхом автоматичного регулювання HVAC і освітлення [2]. Результати показали, що така мікросервісна BEMS-система є більш стійкою і масштабованою, ніж традиційна монолітна: збільшення кількості підключених пристроїв майже не впливає на використання пам'яті, а додавання нових функцій потребує менше часу [2]. Таким чином, мікросервісна архітектура створює основу для відкритих адаптивних систем, які можуть динамічно розширювати функціонал і підлаштовуватися під нові вимоги середовища [1].

Сирі дані від датчиків мікроклімату потребують попередньої обробки, щоб забезпечити надійні рішення системи керування. Основні підходи включають фільтрацію шумів, агрегацію даних та злиття (фузію) даних з різних джерел:

Фільтрація даних. Видалення шумів та нерелевантних точок даних зі стрімкого потоку сенсорної інформації. Наприклад, датчики температури можуть надсилати значення з високою частотою, але не всі зміни є суттєвими для управління [4]. Тому на периферійних вузлах або шлюзах IoT часто реалізують фільтри, які відсікають незначні коливання або використовують ковзне

усереднення/Калманів фільтр для згладжування сигналу. Такий підхід зменшує обсяг переданих в хмару даних і навантаження на мережу [4].

Агрегація даних. Об'єднання даних з кількох датчиків або за певний проміжок часу для отримання усереднених або узагальнених показників. Це може бути розрахунок середнього значення температури по зоні з кількох датчиків, або формування хвилинних інтервалів замість сирих секундних вимірів [4]. Локальна агрегація на шлюзах дозволяє суттєво зменшити кількість повідомлень у мережі, надаючи оглядові показники замість потоків сирих даних [4]. В результаті зменшується кількість запитів до хмарного серверу і економляться обчислювальні ресурси без втрати ключової інформації.

Злиття даних. Інтеграція даних з різних типів датчиків або різних локацій для отримання більш повної картини стану середовища. Злиття може здійснюватися як на рівні особливостей (feature-level), так і на рівні прийняття рішень. Приклад: об'єднання показників температури листя рослин і відносної вологості повітря для розрахунку дефіциту тиску пари (VPD) – важливого індикатора водного балансу рослин [5]. У реальній аеропонній теплиці (Méndez-Guzmán et al., 2022) сенсори листової температури та вологості передають дані на рівень «туману», де обчислюється VPD, який потім зберігається на хмарному сервері ThingSpeak і відображається агроному в мобільному додатку [5]. Інший приклад – обчислення інтегрального показника оптимальності мікроклімату. Rezvani et al. (2020) запропонували об'єднувати виміри температури, вологості і VPD з 20 бездротових датчиків у теплиці в одиничний індекс Optimality Degree (0...1), що показує, наскільки умови наближені до ідеалу для томатів [6]. Така нечітка фузія даних (через функції приналежності) дозволила в реальному часі відстежувати відхилення клімату від норми і виявила проблеми традиційної системи (наприклад, неефективність таймерного керування опаленням/охолодженням) [6].

Багаторівнева обробка (edge-fog-cloud). Часто застосовується розподілена схема, де різні етапи обробки виконуються на різних рівнях системи. Приклад: IoT-архітектура з 4 рівнів для моніторингу теплиці на аеропоніці [5]. На рівні пристроїв датчики збирають показники, далі на проміжному “туманному” рівні здійснюється

первинна обробка – фільтрація шумів, попередній аналіз та навіть комп'ютерний зір (у зазначеній системі – обробка зображень стану листя і коренів) [5]. Оброблені дані передаються до хмарного рівня, де виконуються важчі обчислення: збереження історичних даних, візуалізація трендів і більш складна аналітика, як-от обчислення згаданого VPD або визначення стану системи [5]. Нарешті, на рівні застосунку користувач отримує готову інформацію через веб- або мобільний додаток (наприклад, показники клімату, сигнали тривоги, рекомендації) [5]. Така багат шарова архітектура підвищує оперативність (швидкі реакції на рівні «edge» без затримок мережі) та стійкість (при тимчасових втратах зв'язку хмарні аналітики компенсуються локальною роботою fog-вузла) [5].

Алгоритми МН відіграють ключову роль в адаптивних системах керування мікрокліматом, дозволяючи прогнозувати майбутні стани та приймати оптимальні рішення на основі аналізу великих масивів сенсорних даних. Застосовуються різні підходи, серед яких регресія (прогнозування) параметрів середовища, класифікація станів/подій та адаптивне навчання моделей під нові умови.

Прогнозування (регресія): Для проактивного керування потрібно передбачати зміну кліматичних параметрів. Сучасні дослідження використовують методи глибокого навчання, зокрема рекурентні нейромережі. Наприклад, Thwin et al. розробили модель багатоваріантної багатокрокової LSTM (довготривалої короткочасної пам'яті) для теплиці, що прогнозує температуру і вологість на годину вперед [7]. У систему додатково ввели датчики вологості листя і CO₂ для підвищення точності, а результати прогнозу використовувалися для адаптивного керування обладнанням (вентиляцією в відкритій теплиці) через веб-інтерфейс [7]. Модель досягла високої точності: $R^2 \approx 0.98$ при прогнозуванні температури (RMSE ~ 0.49) і дозволила підтримувати внутрішню температуру ~ 35 °C, коли зовнішня сягала 39–40 °C (вологість залишалась близько 70% – оптимально для рослин) [7]. Інший підхід – поєднання кількох моделей: Tung et al. (2025) запропонували гібридну систему LSTM+SVM, де LSTM відповідає за точний прогноз температури та вологості, перевершуючи класичні методи (для температури RMSE ~ 0.08 при $R^2 = 0.8825$) [7], а простіший RNN виявився найкращим для прогнозу вологості

(RMSE ~ 5.30 , $R^2=0.8187$) [7]. Висока точність прогнозів дозволяє системі завчасно вмикати або вимикати обладнання, пом'якшуючи коливання клімату і економлячи ресурси.

Класифікація та виявлення станів: Окрім кількісного прогнозу, системі важливо розуміти, в якому стані знаходиться середовище або які події відбуваються. Класифікаційні алгоритми застосовуються для розпізнавання ситуацій, як-от критичні стани мікроклімату, погодні умови або наявність людей у приміщенні (для адаптації HVAC). У згаданій гібридній системі Tung et al. паралельно з LSTM використано машину опорних векторів (SVM) для класифікації навколишніх умов (ймовірно, типів погоди чи режимів клімату [7]. SVM досягла точності $\sim 63\%$, перевищивши базові методи (логістична регресія, k-NN) [7]. Хоча 63% може видаватися невисоким показником, така двомодельна система дає комплексний погляд: LSTM безперервно передбачає параметри, а SVM сигналізує про настання певних подій чи станів (наприклад, “небезпечна спека” чи “нормальний режим”) для прийняття якісних рішень у реальному часі [7]. В інших роботах для класифікації застосовувалися нейронні мережі для виявлення присутності людей (щоб перемикаєти режими вентиляції) або класифікатори аномалій для виявлення несправних датчиків/пристроїв, забезпечуючи більш надійне керування системою.

Адаптація моделей (онлайн-навчання): Мікрокліматичні системи працюють у динамічних умовах – зміна сезонів, росту рослин, поведінки користувачів. Тому моделі, навчені на одному наборі даних, з часом можуть втрачати точність. Адаптивні підходи вирішують це шляхом перенавчання моделей на льоту або корекції їх параметрів. Один із методів – онлайн-оцінка параметрів фізичної моделі теплиці: з використанням фільтра Калмана можна безперервно підлаштовувати коефіцієнти моделі, щоб вона відповідала актуальним умовам [researchgate.net](https://www.researchgate.net). Так, у роботі van Henten et al. (2022) застосовано розширений фільтр Калмана (EKF) для ідентифікації змінних параметрів спрощеної моделі клімату – це дозволило підтримувати високу точність моделі протягом довгого періоду без ручного переналаштування [researchgate.net/sciencedirect.com](https://www.researchgate.net/sciencedirect.com). Інший напрям – самоналагоджувані нейронні мережі: моделі, що продовжують навчатися на нових

даних під час експлуатації. Література пропонує архітектури з елементами transfer learning та online learning, де модель зберігає пам'ять про попередні стани і оновлюється при значних відхиленнях [8] Для активного керування перспективними є алгоритми підкріплення (reinforcement learning) – вони можуть в режимі реального часу підбирати оптимальні дії (наприклад, на скільки відкрити вентиляційні люки) на основі зворотного зв'язку про досягнутий результат. Деякі роботи вже продемонстрували успішне застосування глибокого підкріплення для оптимізації клімату теплиць з мінімальними енерговитратами [9] . Таким чином, адаптивні моделі дозволяють системі вчитися з досвіду і залишатися ефективною при зміні умов експлуатації.

Сучасні розумні мікрокліматичні системи поєднують IoT та AI, щоб автоматизувати моніторинг і проактивне налаштування параметрів довкілляmdpi.com. Вони здатні аналізувати поточні показники, робити прогнози та приймати рішення, забезпечуючи сталі оптимальні умови для комфорту людей або росту рослин з мінімальним втручанням людини. Це підвищує ефективність використання ресурсів і сприяє сталому розвитку (наприклад, точному землеробству)mdpi.commdpi.com.

У дослідженнях з адаптивного керування мікрокліматом можна виділити роботи концептуального характеру та прикладні проєкти. Теоретичні роботи часто зосереджені на архітектурі та алгоритмах: наприклад, розробка схем оркестрації мікросервісів, математичних моделей контролера чи методів обробки даних. Такі дослідження перевіряють ідеї в симуляціях або на експериментальних стендах. Приклад – пропозиція тришарової архітектури адаптації для «зелених» будівель, де на основі правил та ML алгоритмів система перебудовує свою структуру в runtime [1]. Ця схема була продемонстрована на стенді CIESOL (Іспанія) для адаптивної автоматизації будівлі, підтверджуючи коректність концепції.

Натомість практичні реалізації фокусуються на інтеграції технологій і отриманні вимірюваних результатів у реальних умовах. Багато таких проєктів з'явилося в агротехнічному секторі: дослідники впроваджують IoT-платформи у діючих теплицях чи на полях. Система IRRISENS (2020) була розгорнута на

справжніх фермах з різними культурами і працювала впродовж сезону, збираючи дані та підтримуючи зрошення за допомогою хмарних сервісів [3]. Adaptive Greenhouse system від Thwin et al. (2024) випробувана у робочій теплиці в Таїланді – впродовж експериментів модель керувала вентиляцією і підтримувала стабільний клімат, що підтверджує ефективність підходу в польових умовах [7]. Такі практичні кейси часто висвітлюють інженерні виклики (надійність бездротової передачі, енергоживлення датчиків, кібербезпека) і показують, яких показників вдається досягти (напр., економія води, підвищення врожайності чи зниження енергоспоживання).

1.4 Визначення проблем та методів їх рішення

При створенні програми на мікросервісній архітектурі може виникнути такі проблеми:

Зі збільшенням кількості сервісів зв'язок між ними може стати завантаженим та складним, що призведе до збільшення затримок. Для вирішення цієї проблеми буде використано полегшені протоколи зв'язку, наприклад HTTP, або черги повідомлень. Це допоможе мінімізувати навантаження та витрати.

Підтримка узгодженості даних. Забезпечення єдності в мікросервісах може бути складною задачею через децентралізованість додатку. Тобто кожен сервіс створює свій зв'язок з власною базою даних. Для уникнення колізій та перезаписів буде використано асинхронний обмін повідомленнями для передачі оновлених даних між сервісами. В кожному сервісі створено контекст бази даних який стежить за своїми об'єктами.

Розгортання. Процес розгортання може бути дуже затратним, тому для цього використовують технологію контейнеризації. Використовуючи Docker можна запакувати кожен мікросервіс та його залежності окремо та вивести в хмару. Після цього використовуючи файли конфігурації запусити їх на віддаленому пристрої.

Виявлення сервісів і балансування. Для балансування та переадресації на кінцеві точки сервісів буде використано Ocelot. Завдяки цій бібліотеці можна

реалізувати ці механізми та авторизацію за Bearer-токеном.

Моніторинг активності мікросервісів. Для цього буде використовуватись механізм логування вбудований в фреймворк ASP.NET та додаткова конфігурація за допомогою бібліотеки NLog.

1.5 Постановка задачі

У сучасних умовах підвищеної уваги до енергоефективності, якості навколишнього середовища та автоматизації критично важливих процесів, виникає потреба в побудові гнучких, адаптивних та масштабованих систем керування мікрокліматом. Традиційні монолітні системи не забезпечують необхідної адаптивності, швидкості обробки великих обсягів сенсорних даних та інтеграції з хмарними або edge-обчисленнями.

Метою даної бакалаврської роботи є розробка мікросервісної системи адаптивного керування мікрокліматом, яка буде здатна:

- збирати та аналізувати дані з датчиків температури, вологості, CO₂ та освітленості;
- приймати автоматизовані рішення щодо регулювання відповідних параметрів середовища.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- Провести аналіз предметної області, сучасних підходів до управління мікрокліматом, IoT-технологій і мікросервісної архітектури.
- Визначити архітектуру системи та функціональні модулі (сервіси збору даних, обробки, керування та взаємодії з виконавчими пристроями).
- Реалізувати набір мікросервісів для збору сенсорних даних, збереження інформації, обчислення контрольних дій та взаємодії з пристроями керування.
- Реалізувати RESTful API для взаємодії з клієнтськими додатками.
- Застосувати базові методи аналізу даних для оцінки поточного стану середовища та формування рішень.
- Провести тестування системи у симуляційному або реальному середовищі

та оцінити ефективність прийнятих рішень.

Результатом виконання роботи має стати прототип мікросервісної системи, придатний для розгортання у контейнеризованому середовищі (наприклад, Docker) та здатний до подальшого розвитку із залученням методів штучного інтелекту для прогнозування стану мікроклімату.

2 МІКРОСЕРВІСНА АРХІТЕКТУРА ТА СІТІ-ФЕРМЕРСТВО

2.1 Основні характеристики мікросервісів

Мікросервісна архітектура – це підхід до розробки програмного забезпечення, при якому додаток розбивається на набір невеликих, незалежних сервісів (мікросервісів), кожен з яких виконує одну конкретну бізнес-функцію та спілкується з іншими через чітко визначені API (наприклад, HTTP/REST, gRPC, Messaging).

До основних характеристик мікросервісів можна віднести:

Незалежність – кожен мікросервіс розробляється, розгортається та масштабується окремо.

Спеціалізація – один сервіс відповідає за одну функцію (наприклад, аутентифікація, оплата, нотифікації).

Розподілена система – сервіси можуть бути розгорнуті на різних серверах або контейнерах (Docker, Kubernetes).

Слабка зв'язність – сервіси спілкуються через мережу, а не через спільну базу даних.

Автономність – кожен мікросервіс має свою БД або схему (наприклад, PostgreSQL для одного, MongoDB для іншого).

Перевагами мікросервісів є:

Гнучкість – можна використовувати різні технології для різних сервісів.

Масштабованість – можна масштабувати окремі компоненти, а не весь додаток.

Надійність – падіння одного сервісу не "валить" всю систему.

Швидкість розробки – різні команди можуть працювати паралельно.

Полегшене тестування та деплой – оновлення одного сервісу не вимагає перезапуску всього додатка.

Серед недоліків можна виділити:

Складність управління – потрібні інструменти для оркестрації (Kubernetes),

моніторингу (Prometheus, Grafana), логування (ELK).

Мережеві затримки – комунікація між сервісами через API може бути повільнішою, ніж у моноліті.

Складність транзакцій – важко забезпечити ACID у розподілених системах (вирішують через Saga, CQRS, Event Sourcing).

Використовувати мікросервіси доцільно при таких обставинах:

Великі та складні додатки (наприклад, Uber, Netflix, Airbnb).

Системи, що швидко масштабуються (окремі модулі можуть обробляти більше навантаження).

Команди розробників розподілені – кожна команда працює над своїм сервісом.

Різні технологічні стеки – де різні частини системи вимагають різних підходів (наприклад, AI-моделі + веб-інтерфейс).

Висока доступність – якщо критично, щоб система залишалася живою при збоях окремих компонентів.

Проте не варто використовувати мікросервіси, коли:

Невеликі проекти – додаткові накладні витрати не виправдані.

Моноліт простіший – якщо додаток не планує масштабуватися.

Немає досвіду з DevOps – розгортання та управління мікросервісами вимагає кваліфікованих фахівців.

Отже, мікросервіси підходять для складних, масштабованих систем, але вимагають додаткових ресурсів на підтримку. Для стартапів або простих додатків часто краще підходить моноліт або модульний моноліт.

2.2 Сіті-фермерство

Мікрозелене виробництво стає все більш популярним серед малих і міських фермерів. Мікрозелень зазвичай потребує обмежених початкових інвестицій, використовує мало ресурсів і місця, має швидкий оборот, її можна вирощувати цілий рік за допомогою різноманітних методів і, як правило, дуже прибуткова.

Оскільки мікрозелень збирають через один-два тижні після проростання насіння, обсяг виробництва можна швидко регулювати, щоб задовольнити попит і запобігти відходам. Мікрозелень також є делікатною і легко пошкоджується, тому важливо захистити її від небажаних факторів навколишнього середовища, таких як дощ і вітер [14].

Устаткування міської ферми використовує сучасні пристрої контролю клімату, апаратні засоби до яких включають контролери, датчики, перемикачі, кліматична та опалювальна техніка, лічильники, тощо. Через те, що промисловість дислокується в приміщенні, є потреба у власному кліматі: певна вологість і температура для різних культур, день і ніч, штучне освітлення, штучний вітер і вентиляція.

Зазвичай архітектура кліматичної установки являє собою конструктивні елементи систем опалення, вентиляції та кондиціонування, в тому числі радіатор опалення, вентилятор повітря та кондиціонер, що складається з випарника, компресора, конденсатора і ресивера. Керування та управління кліматичною установкою здійснює відповідна система. Головними елементами системи є вхідні датчики, блок управління та виконавчі пристрої. Вхідні датчики вимірюють відповідні фізичні параметри і перетворюють їх в сигнали. До складу вхідних датчиків системи управління відносяться датчики температури зовнішнього повітря, рівня сонячного випромінювання (фотодіод), вихідної температури, потенціометри заслінок, температури випарника, тиску в системі кондиціонування. Кількість датчиків вихідної температури визначається конструкцією системи клімат-контролю.

Системи опалення, вентиляції, зволоження складаються з ряду підсистем, кожна з яких може мати нелінійні характеристики, що змінюються з часом. Причому параметри динамічного стану зазвичай змінюються в залежності від погоди і навколишніх умов. Але використання традиційних схем управління для такої технології та систем управління з великою кількістю параметрів, що регулюються, є неефективним.

Для управління системою терморегуляції (опалення, вентиляції та кондиціонування повітря) на об'єкті може використовуватися класична система

GDS-регулювання, хоча при зміні параметрів об'єкта може впливати і на систему ОВК [15]. Впровадження додаткових елементів управління має певні недоліки, такі як незручна настройка параметрів, перешкоди, незначні відхилення від заданих значень.

Конденсовану електричну систему опалення можна запрограмувати на тижневий або місячний цикл опалення. Контроль температури в приміщенні дещо складніший. Термостат повинен мати можливість не тільки управляти нагрівом повітря і швидкістю обертання вентилятора, але і оптимальним алгоритмом регулювання температури повітряного охолодження системи кондиціонування повітря.

На ріст рослин, цвітіння, зав'язування плодів і розмір позитивно впливає активний контроль клімату, що призводить до більшої та кращої врожайності та до раннього початку виробництва рослин [16].

Для оптимізації температурних умов експлуатації теплиці застосовують наявні теплові моделі та інструменти моделювання [17]. У таких випадках можуть використовуватись теплиці, інтегровані з кількома системами, які забезпечують сприятливий мікроклімат для вирощування сільськогосподарських культур в умовах екстремального, тропічного та субтропічного клімату.

Також існують стратегії енергоефективного контролю та найсучаснішого енергетичного моделювання для теплиць. Ці стратегії використовуються для підвищення енергоефективності при роботі з контролем парникових газів шляхом узагальнення досліджень щодо стратегій енергоефективної роботи, контролю ключових параметрів теплиць, мережі датчиків і систем моніторингу, а також різноманітних алгоритмів керування.

В сучасному світі основними компонентами системи моніторингу та контролю є датчики. Ключовою необхідністю для захисту культур від несприятливих середовищ окрім моніторингу та контролю мікрокліматичного середовища в теплиці є і програмні методи зондування. Важливе значення також має правильне розташування датчиків та їх захист від агресивних факторів навколишнього середовища [18].

В деяких випадках важливо кількісно вивчити мікроклімат у теплиці, щоб відповідно оптимізувати схему переміщення рослин, тому що значний рівень шуму впливає на зріст рослин. Для оптимізації відстані та частоти переміщення горщика було запропоновано підхід комп'ютерного моделювання для моделювання мікроклімату в теплиці. Отримана схема переміщення може усунути понад 90% дисперсії мікроклімату, але може заощадити понад 95% зусиль переміщення порівняно з безперервним рухом [19].

Штучний інтелект (ШІ) досяг прориву в багатьох сферах, але поки що не в садівництві. Метою міжнародного конкурсу «автономних теплиць» було поєднання садівничого досвіду із штучним інтелектом, щоб зробити прорив у виробництві свіжої їжі з меншими ресурсами. П'ять міжнародних команд, що склалися з науковців, професіоналів і студентів з різним досвідом у садівництві та штучному інтелекті, брали участь в експерименті з тепличного вирощування. Кожна команда мала сучасну теплицю площею 96 м² для дистанційного вирощування врожаю огірків протягом 4-х місяців. Кожен відсік був обладнаний стандартними приводами (опалення, вентиляція, екранування, освітлення, туман, подача CO₂, подача води та поживних речовин). Задані контрольні значення дистанційно визначали команди за допомогою власних алгоритмів штучного інтелекту (ШІ). Виконавчими механізмами керував технологічний комп'ютер. Різні датчики постійно збирали вимірювання. Задані значення та вимірювання обмінювалися через цифровий інтерфейс. Досягнення у відсіках, керованих штучним інтелектом, порівнювали з еталонним зразком, що керувався вручну. Загалом ШІ добре справлявся з керуванням теплицею.

Розробка та впровадження сіті-фермерського підприємства з передовими системами контролю клімату сприяє стабілізації виробництва, підвищенню економічної ефективності та збереженню ресурсів. Попри певні недоліки, їх можна мінімізувати завдяки сучасним інтелектуальним технологіям:

– Сенсори та пристрої IoT, встановлені в різних зонах ферми, забезпечують моніторинг і керування кліматичними параметрами в реальному часі [20], що дозволяє оперативно реагувати на проблеми та зменшувати ризики втрат [21].

– Системи ШІ аналізують дані сенсорів і прогнозують оптимальні умови для росту рослин, що сприяє ефективному використанню ресурсів.

– Автоматизовані системи й роботи підвищують точність керування та знижують залежність від людського фактора.

– Технологія блокчейн забезпечує надійне та прозоре управління даними про клімат і виробництво.

Мікрозелень, молода зелень, вирощена з насіння овочів, трав або зернових, за останні кілька років набирає популярності серед малих і міських фермерів (рис. 2.1). Мікрозелень відносно легко вирощувати, її можна вирощувати в приміщенні з обмеженими ресурсами, і вона має короткий цикл росту.

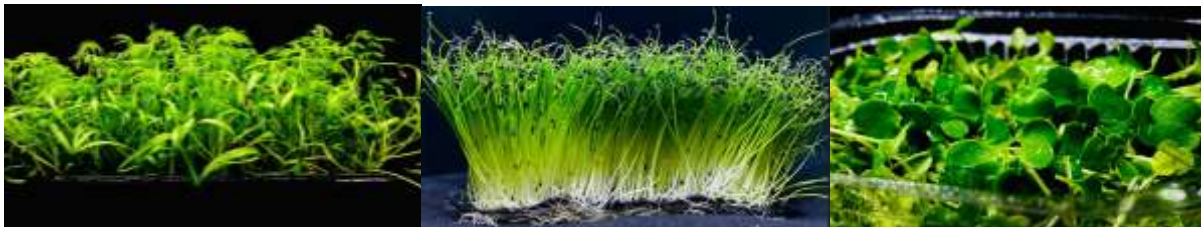


Рисунок 2.1– Мікрозелень фенхеля, смітцевої цибулі та соняшника нашої сіті-ферми

Все більше останніх досліджень підтверджують користь мікрозелені для здоров'я завдяки високій щільності вітамінів і поживних речовин. Вони популярні серед споживачів, які піклуються про своє здоров'я, і кухарів, які використовують мікрозелень як барвисті та ароматні гарніри (рис. 2.2).



Рисунок 2.2 – Квітка пеларгонії як барвистий гарнір

Мікрозелень вирощують у закритих приміщеннях із використанням різних матеріалів і методів. Основні компоненти — насіння, середовище для росту, світло,

вода та лотки. Застосовують як ґрунтові, так і безґрунтові методи, зокрема гідропоніку (рис. 2.3).



Рисунок 2.3 – Вирощування мікрозелені в закритому приміщенні

Сорти мікрозелені мають різну тривалість росту, тому суміші слід формувати з видів із подібними циклами. Її вирощують при природному або штучному освітленні, зазвичай при температурі близько 24 °С. Для запобігання цвілі важлива вентиляція — циркуляція повітря, наприклад, за допомогою вентилятора. Середовище та коріння мають залишатися вологими для кращого проростання насіння.

Вирощування мікрозелені в закритому приміщенні потребує точного клімат-контролю. Система регулює температуру (близько 24 °С), вологість, освітлення та циркуляцію повітря, що запобігає цвілі та сприяє здоровому росту. Штучне освітлення забезпечує необхідний рівень світла, а оптимальні умови розвитку підвищують врожайність.

3 ПРОЄКТУВАННЯ МІКРОСЕРВІСНОЇ СИСТЕМИ АДАПТИВНОГО КЕРУВАННЯ МІКРОКЛІМАТОМ

У мікросервісній системі адаптивного керування мікрокліматом кожен мікросервіс є незалежним програмним компонентом, який виконує чітко визначену функцію, таку як вентиляція, опалення чи контроль освітлення. Ці сервіси побудовані за принципом автономності: вони мають власну логіку, окремий API та не залежать від внутрішньої реалізації інших частин системи. Їх спеціалізація дозволяє зосередитись на одній задачі — наприклад, контроль температури, моніторинг вологості або обробка показників CO₂. Усі компоненти системи взаємодіють між собою через легкі протоколи (REST, MQTT, WebSocket), що забезпечує гнучкість інтеграції та обміну даними.

Завдяки мікросервісному підходу, така система легко масштабується — нові сервіси можуть бути додані без зупинки основної інфраструктури, а існуючі — оновлені чи замінені незалежно. Адаптивність системи проявляється у здатності кожного сервісу динамічно реагувати на зміну зовнішніх умов, таких як температура повітря або присутність людей. Наприклад, сервіс температурного контролю регулює роботу обігрівачів, отримуючи актуальні показники з термодатчиків, тоді як вологісний модуль відповідає за зволоження або осушення повітря, ґрунтуючись на даних вологості. Водночас інший мікросервіс може контролювати рівень CO₂, автоматично активуючи вентиляцію при перевищенні допустимого порогу.

Ще один важливий компонент — модуль енергоефективності — аналізує споживання енергії та адаптує режим роботи пристроїв з урахуванням розкладу або присутності користувачів. Інтеграція з IoT дозволяє системі керувати «розумними» приладами — наприклад, відкривати чи закривати жалюзі для регулювання рівня освітлення та теплового навантаження. Перевагами такої архітектури є гнучкість у розробці, висока відмовостійкість (помилка одного сервісу не впливає на роботу інших), простота масштабування та можливість використання різних технологій у

межах однієї системи.

У результаті, мікросервісна архітектура забезпечує адаптивне, надійне та ефективне керування мікрокліматом у розумних будівлях, теплицях або промислових об'єктах, де важлива точна реакція на зміни середовища.

3.1 План розробки мікросервісної системи

Детальний план розробки мікросервісної системи адаптивного керування мікрокліматом охоплює всі ключові етапи проєктування, реалізації та впровадження, що дозволяє системно підійти до створення інтелектуальної, гнучкої та масштабованої системи з високим ступенем адаптивності. На початковому етапі проводиться аналіз функціональних вимог, зокрема визначення параметрів середовища, які необхідно контролювати: температура, вологість, вміст CO₂ та рівень освітлення. Паралельно встановлюються типи виконавчих пристроїв, серед яких обігрівачі, кондиціонери, зволожувачі, вентиляційні установки та освітлювальні системи.

Далі обирається відповідний архітектурний підхід, і в цьому контексті мікросервісна архітектура є найдоцільнішою, адже передбачає незалежність окремих компонентів системи. Визначаються основні функціональні блоки: сервіси збору даних, зберігання, аналітики, керування пристроями, API та моніторингу. Для візуалізації взаємодій між ними створюється узагальнена схема потоків даних з урахуванням типів комунікації — як синхронної (REST), так і асинхронної (черги подій, MQTT).

Проєктування мікросервісів передбачає реалізацію спеціалізованих компонентів: температурного сервісу, який обробляє дані з термодатчиків і регулює роботу обігрівачів або кондиціонерів; сервісу вологості, що взаємодіє з відповідними сенсорами та керує зволожувачами або осушувачами; модуля CO₂, який забезпечує моніторинг вуглекислого газу та активує вентиляцію при перевищенні порогу; сервісу освітлення, що регулює штучне освітлення або розумні штори; аналітичного сервісу, який агрегує дані, виявляє аномалії та виконує

прогнозування на основі алгоритмів машинного навчання; а також контролера пристроїв, який виступає посередником між аналітикою та виконавцями.

Комунікація між сервісами організовується через стандартизовані REST API або gRPC-інтерфейси. Для обміну даними застосовуються протоколи HTTP/HTTPS, MQTT (для IoT-пристроїв), а також RabbitMQ або Kafka для асинхронного обміну подіями. Захист API забезпечується за допомогою JWT-токенів, використовується шифрування HTTPS і механізми контролю доступу на основі ролей (RBAC).

На рівні обробки даних система здійснює збір інформації в реальному часі з подальшою фільтрацією та нормалізацією. Історичні дані зберігаються у time-series базах (InfluxDB, TimescaleDB), тоді як метадані — у реляційній базі PostgreSQL. Для прийняття рішень застосовуються rule-based алгоритми, а також моделі машинного навчання, зокрема LSTM, регресійні та класифікаційні методи.

Інтеграція з сенсорною та виконавчою інфраструктурою здійснюється через MQTT-брокери та стандартні протоколи, як-от Modbus, Zigbee або Wi-Fi. У керуванні пристроями використовуються HTTP-запити, GPIO або MQTT, причому час реакції не перевищує двох секунд.

Масштабованість і надійність системи досягається через контейнеризацію всіх сервісів у Docker, з можливістю централізованого оркестрування за допомогою Docker Compose або Kubernetes. Для моніторингу використовуються Prometheus і Grafana, а для виявлення збоїв — система алертів. Логування здійснюється централізовано через NLog або стек ELK.

На завершальному етапі проводиться тестування — як модульне, так і інтеграційне, а також навантажувальне за допомогою інструментів типу JMeter. Система розгортається через CI/CD пайплайни з розмежуванням середовищ для тестування та продуктивної експлуатації. Такий підхід гарантує адаптивність, надійність і зручну інтеграцію розробленої системи в існуючі інфраструктури розумного середовища.

3.2 Визначення вимог до системи

Розробка мікросервісної системи адаптивного керування мікрокліматом передбачає чітке формулювання функціональних та нефункціональних вимог, які визначають архітектуру, логіку роботи та очікувану поведінку системи в реальних умовах експлуатації. На функціональному рівні система повинна забезпечувати автоматизований збір даних із сенсорів температури, вологості, CO₂ та освітленості, а також зберігати ці дані у базі даних із точною часовою міткою для подальшого аналізу. Крім накопичення інформації, система повинна бути здатною на основі отриманих значень приймати рішення щодо вмикання або вимикання керуваних пристроїв, зокрема вентиляційного обладнання, обігрівачів чи інших виконавчих механізмів.

Користувач повинен мати змогу взаємодіяти з системою через API, зокрема отримувати поточний стан мікроклімату, переглядати значення сенсорів у реальному часі, а також аналізувати динаміку показників за визначений часовий період. Це реалізується за допомогою клієнтського застосунку або веб-інтерфейсу, який дозволяє доступ до історичних даних і результатів автоматичних рішень.

Щодо нефункціональних вимог, система повинна гарантувати обробку даних із мінімальною затримкою — не більше двох секунд від моменту надходження вимірювання до прийняття відповідного керувального рішення. Для досягнення високої доступності та відмовостійкості всі мікросервіси розробляються як ізольовані контейнери, які запускаються у середовищі Docker і можуть бути незалежно оновлені або замінені без зупинки всієї системи. Важливо, щоб при виході з ладу одного з компонентів (наприклад, сервісу логування або збереження даних) інші частини продовжували виконувати свої функції, зокрема реагування на критичні зміни параметрів мікроклімату.

Ілюстрацією до вищезазначених вимог може слугувати типовий сценарій використання. Наприклад, користувач відкриває мобільний застосунок для перегляду температурного профілю в теплиці за останні 24 години. Якщо система виявляє, що температура перевищує встановлений пороговий рівень, вона автоматично генерує керувальну команду та вмикає систему вентиляції. Усе це відбувається без втручання користувача, але з можливістю контролю та перегляду

рішень у реальному часі. Такий підхід забезпечує як автоматизацію управління, так і прозорість процесів, що є критично важливими у розумних середовищах.

3.3 Архітектура системи

Архітектура системи адаптивного керування мікрокліматом базується на мікросервісному підході, що забезпечує модульність, гнучкість, масштабованість і високу відмовостійкість. Кожен сервіс виконує чітко визначену функцію, що дозволяє розробляти, розгортати та оновлювати окремі компоненти незалежно один від одного. Завдяки такому підходу система легко адаптується до змін вимог або розширення функціоналу, що є особливо важливим в умовах динамічного середовища.

У центрі архітектури знаходиться *Sensor Ingestion Service*, який відповідає за прийом і первинну обробку даних з датчиків температури, вологості, CO₂ та освітленості. Ці дані у форматі потоків або повідомлень передаються далі до *Analytics Service*, що виконує аналіз показників, визначає відхилення від нормативних значень, виявляє аномалії та формує рішення щодо керування середовищем.

Результати обчислень передаються до *Actuator Control Service*, який безпосередньо взаємодіє з виконавчими пристроями, такими як вентилятори, обігрівачі, освітлення або зволожувачі, і надсилає команди на їх увімкнення або вимкнення відповідно до поточного стану середовища та заданої логіки. Паралельно всі вхідні та оброблені дані фіксуються у *Storage Service*, що забезпечує збереження значень сенсорів та дій системи з часовими мітками в базі даних. Це створює основу для ретроспективного аналізу, трендової аналітики та навчання моделей прогнозування.

Усі зовнішні запити до системи, зокрема від користувацьких додатків або інших сервісів, проходять через *API Gateway*, який виконує функції маршрутизації, балансування навантаження, автентифікації та авторизації доступу до відповідних мікросервісів. Це дозволяє централізовано контролювати точки входу до системи та

забезпечує безпеку комунікацій.

Для підтримки працездатності та діагностики функціонування компонентів у режимі реального часу реалізовано Monitoring Service, який збирає технічні метрики, фіксує журнали подій, відстежує навантаження та дозволяє оперативно реагувати на збої. Завдяки цій архітектурі система є не лише функціонально гнучкою, але й готовою до масштабування, інтеграції нових сервісів або пристроїв без порушення стабільності роботи (рис. 3.1).

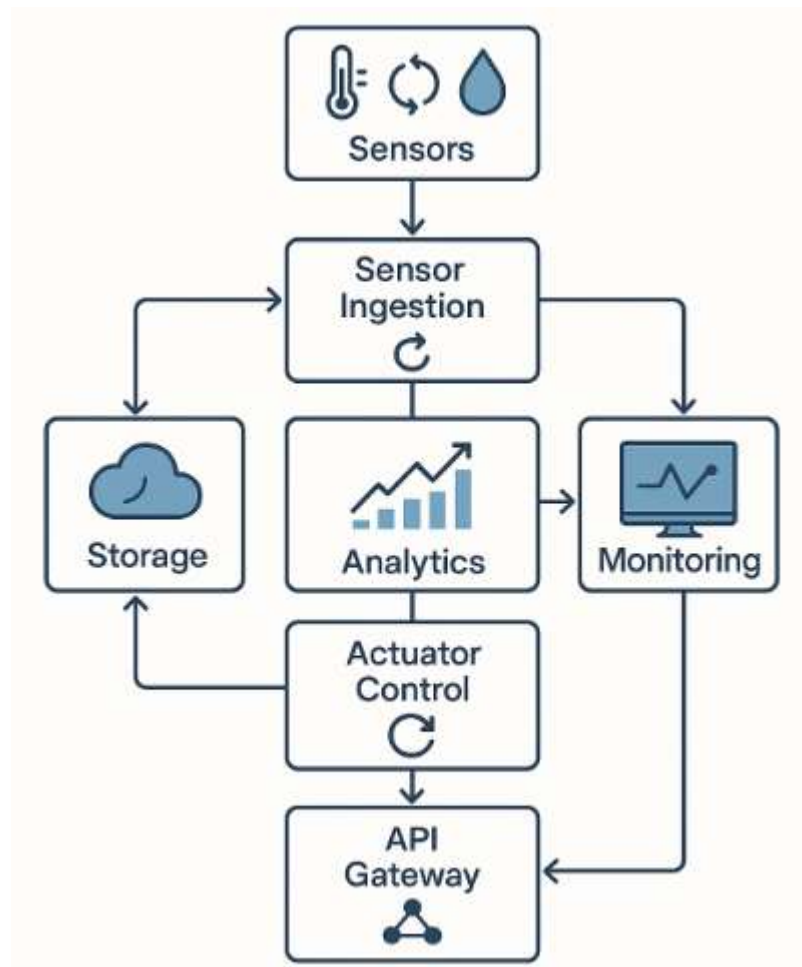


Рисунок 3.1 – Архітектура системи адаптивного керування мікрокліматом

Sensors — відповідає за збирання первинних даних про температуру, вологість, CO₂ та освітленість.

Sensor Ingestion — приймає потоки даних від сенсорів і переспрямовує їх до інших сервісів.

Storage — зберігає дані, що надходять, у базі даних (у тому числі для

ретроспективного аналізу).

Analytics — обробляє показники, виявляє відхилення, виконує аналіз та формує рішення.

Actuator Control — керує виконавчими пристроями на основі рішень аналітики.

API Gateway — забезпечує централізований доступ до функціоналу системи ззовні.

Monitoring — здійснює спостереження за станом системи і її компонентів.

Стрілки між блоками відображають потоки даних та команд, зокрема: від сенсорів — до ingestion; від ingestion — до storage, analytics і monitoring; від analytics — до actuator control і monitoring; від actuator control — до API gateway; а також зворотний зв'язок від API gateway до actuator control і monitoring.

3.4 Проектування мікросервісів

Проектування мікросервісної системи адаптивного керування мікрокліматом передбачає створення окремих незалежних сервісів, кожен з яких виконує специфічну функцію в загальній структурі керування. Такий підхід дозволяє досягти високої гнучкості системи, спрощує її масштабування, тестування та супровід, а також дає змогу поетапно розширювати функціонал.

Ключову роль у взаємодії з фізичними пристроями відіграє Sensor Ingestion Service, який відповідає за прийом MQTT-повідомлень з IoT-сенсорів, що передають значення температури, вологості, освітлення та рівня CO₂. Цей сервіс виконує попередню обробку даних, зокрема нормалізацію значень і перевірку коректності структури повідомлень перед передачею їх на подальший аналіз і зберігання.

Отримані дані передаються до Storage Service, який реалізовано на базі реляційної бази даних PostgreSQL. Усі вимірювання зберігаються з точною часовою прив'язкою, що дозволяє відстежувати зміни параметрів у динаміці, проводити ретроспективний аналіз і використовувати історичні дані для навчання прогнозних моделей.

Центральним елементом інтелектуальної обробки є Analytics Service, який виконує обчислення середніх значень, виявляє аномальні стани та здійснює прогнозування кліматичних параметрів за допомогою моделей на основі довготривалої короткочасної пам'яті (LSTM). Отримані аналітичні висновки формують основу для прийняття рішень про зміну режимів роботи системи.

Прийняті рішення передаються до Actuator Control Service, який формує команди для пристроїв управління середовищем — зокрема, систем опалення, вентиляції, зволоження або освітлення. Цей сервіс відповідає за взаємодію з реальними виконавчими механізмами, забезпечуючи відповідність їхньої поведінки поточним умовам і рекомендаціям аналітичного модуля.

Для координації взаємодії між зовнішніми клієнтами та мікросервісами використовується API Gateway, побудований на базі Ocelot. Він відповідає за авторизацію запитів через JWT-токени, виконує маршрутизацію до відповідних REST API сервісів, а також забезпечує балансування навантаження між екземплярами сервісів у разі масштабування.

З метою контролю стабільності роботи всієї системи впроваджено Monitoring Service, який інтегрує засоби спостереження Prometheus і Grafana. Цей сервіс дозволяє відстежувати технічні метрики кожного компонента, створювати графіки продуктивності, виявляти відхилення та оперативно реагувати на критичні події за допомогою системи алертів.

Разом усі ці мікросервіси формують взаємопов'язану, адаптивну і масштабовану систему, здатну динамічно реагувати на зміни мікроклімату та забезпечувати ефективне керування умовами середовища у приміщенні.

На рисунку 3.2 представлено діаграму процесу проєктування та реалізації мікросервісної системи адаптивного керування мікрокліматом, яка відображає поетапну послідовність дій — від початку проєкту до його завершення. Діаграма розділена на два вертикальні блоки, пов'язані між собою стрілкою, що демонструє логічний перехід між етапами.

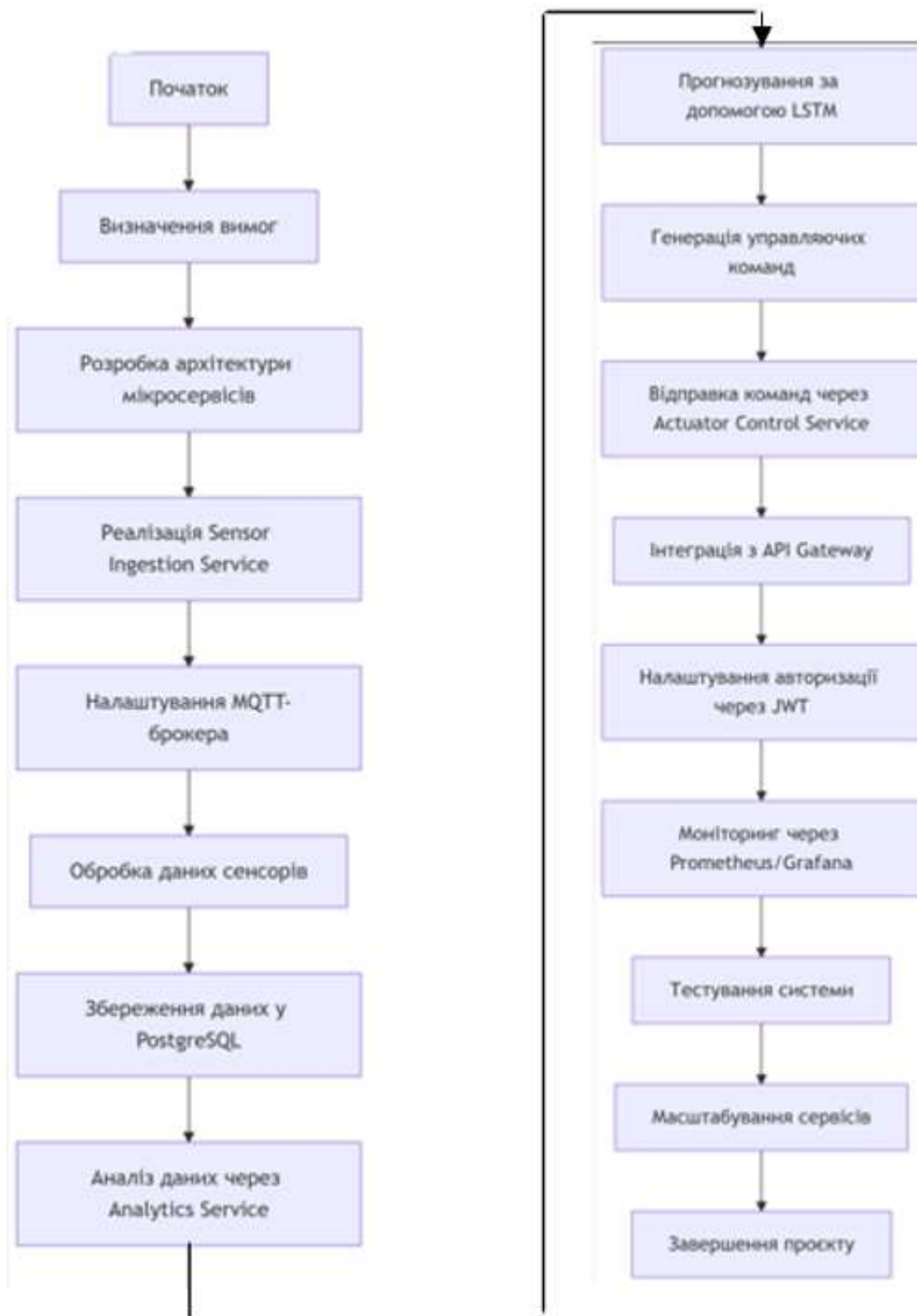


Рисунок 3.2 – Схема процесу проєктування та реалізації мікросервісної системи адаптивного керування мікрокліматом

У лівій частині діаграми зображено етапи початкової розробки: визначення вимог, проєктування архітектури мікросервісів, реалізація сервісу прийому даних Sensor Ingestion, налаштування MQTT-брокера, обробка сенсорних даних, збереження даних у базі PostgreSQL, а також аналітика за допомогою Analytics

Service.

Права частина діаграми ілюструє наступні дії після аналітики: прогнозування параметрів мікроклімату з використанням моделі LSTM, генерація керувальних команд, відправка команд через Actuator Control Service, інтеграція з API Gateway, налаштування авторизації через JWT, моніторинг роботи системи за допомогою Prometheus та Grafana, тестування системи, масштабування сервісів і завершення проєкту.

Схема чітко і послідовно демонструє повний життєвий цикл створення мікросервісної системи — від формування концепції до реалізації, тестування та впровадження.

На рисунку 3.3 представлено спрощену структурну діаграму проєктування мікросервісної системи адаптивного керування мікрокліматом, яка відображає основні компоненти системи та напрямки їхньої взаємодії.



Рисунок 3.3 – Структурна діаграма проєктування мікросервісної системи адаптивного керування мікрокліматом

Sensor Ingestion (MQTT → Валідація). Цей мікросервіс отримує дані з сенсорів (температури, вологості, CO₂ тощо) через протокол MQTT. Після отримання відбувається валідація даних для перевірки їхньої коректності перед передачею далі

в систему.

Storage Service (PostgreSQL). Прийняті та перевірені дані передаються до служби збереження, яка працює на базі реляційної СУБД PostgreSQL. Тут зберігаються як поточні значення сенсорів, так і історичні записи для подальшого аналізу.

Analytics Service (LSTM, аналіз). Цей компонент аналізує зібрані дані, зокрема, виконує прогнозування кліматичних параметрів за допомогою алгоритмів машинного навчання (наприклад, LSTM). На основі аналізу формується рішення про дії щодо зміни мікроклімату.

API Gateway (Ocelot + JWT). API Gateway виконує функцію шлюзу для зовнішніх запитів. Він здійснює авторизацію користувачів (зокрема через JWT) і маршрутизує запити до відповідних внутрішніх сервісів.

Actuator Control (Виконавчі пристрої). На основі прогнозу або рекомендацій з аналітики ця служба формує команди керування для виконавчих пристроїв — систем вентиляції, освітлення, підігріву тощо.

Monitoring Service (Prometheus/Grafana). Відповідає за моніторинг усіх сервісів, збір метрик (Prometheus) та візуалізацію показників (Grafana). Також слідкує за станом виконавчих пристроїв і повідомляє про збої чи аномалії.

Потік починається з отримання даних від сенсорів. Дані зберігаються, передаються на аналітику, а звіди — на виконавчі пристрої. API Gateway забезпечує зовнішній доступ до системи, включаючи аналітичні сервіси. Monitoring Service контролює стабільність усіх блоків і пристроїв.

4 ПРОЄКТУВАННЯ МОДУЛІВ АНАЛІЗУ ДАНИХ

Аналітика даних у системі адаптивного керування мікрокліматом є ключовим елементом, що забезпечує інтелектуальне прийняття рішень на основі сенсорної інформації в реальному часі та історичних трендів. Основна мета аналітики — не просто обробити вхідні дані, а виявити закономірності, передбачити зміни параметрів середовища і сформувані ефективні керувальні дії.

4.1 Основні функції аналітики

Попередня обробка та очищення даних. Система отримує значення температури, вологості, рівня CO₂ та освітленості у вигляді потоків MQTT-повідомлень. Дані проходять фільтрацію (наприклад, за допомогою ковзного середнього чи фільтра Калмана) для усунення шумів, пропусків або аномальних значень.

Агрегація і статистичний аналіз. Оброблені показники агрегуються у вигляді середніх, мінімальних, максимальних або дисперсійних значень на заданих інтервалах часу (наприклад, щохвилинно чи щогодинно). Це дозволяє зменшити обсяг даних і виявити відхилення від нормальних умов.

Виявлення аномалій. Аналітичний сервіс автоматично ідентифікує відхилення від типових значень на основі порогів або методів класифікації. Наприклад, може виявляти надмірний рівень вологості або CO₂, який вказує на порушення вентиляції або присутність людей у приміщенні.

Прогнозування кліматичних параметрів. Для проактивного керування середовищем використовується прогнозування температури, вологості або CO₂ на основі часових рядів за допомогою моделі LSTM (Long Short-Term Memory) — типу рекурентної нейронної мережі, яка добре справляється з динамічними змінами у часі. Наприклад, модель може передбачити підвищення температури через 30 хвилин і завчасно активувати вентиляцію або зменшити освітлення.

Формування рекомендацій. На основі аналізу та прогнозу система генерує керувальні стратегії у вигляді команд для виконавчих пристроїв. Наприклад, у разі перевищення температурного порогу активується команда на ввімкнення охолодження або відкриття жалюзі.

Оцінка ефективності управління. Після виконання команд система продовжує моніторинг змін у параметрах середовища, фіксуючи, чи було досягнуто бажаного ефекту. Це дозволяє аналізувати ефективність обраної дії та використовувати зворотний зв'язок для навчання моделей у майбутньому.

На рисунку 4.1 наведено зображення аналітичного конвеєра, яке демонструє поетапну обробку сенсорних даних у системі адаптивного керування мікрокліматом:



Рисунок 4.1 – Схема аналітичного конвеєра

– Вхідні дані датчиків — надходять у вигляді потоків від сенсорів температури, вологості, CO₂ тощо.

– Попередня обробка — включає фільтрацію шумів, згладжування та інтерполяцію для забезпечення цілісності й коректності даних.

- Обчислення показників — визначаються статистичні характеристики: середнє, дисперсія, тренди.
- Виявлення аномалій — відбувається через порогову логіку або класифікатори, які сигналізують про відхилення.
- LSTM-прогнозування — модель на основі рекурентної нейромережі передбачає майбутній стан середовища (наприклад, температуру через 30 хв).
- Генерація рекомендацій — на основі аналізу й прогнозу формуються дії щодо систем опалення, вентиляції, освітлення.
- Команди — результати аналітики перетворюються в керувальні сигнали для виконавчих пристроїв.

4.2 LSTM-модель для прогнозування параметрів мікроклімату

LSTM (Long Short-Term Memory) — це тип рекурентної нейронної мережі, яка забезпечує ефективне навчання на часових рядах, враховуючи довготривалі залежності між послідовними значеннями. У контексті системи адаптивного керування мікрокліматом така модель застосовується для прогнозування майбутніх параметрів середовища, зокрема температури, вологості, рівня CO₂ або освітленості, на основі історичних вимірювань.

На зображенні 4.2 представлено схематичну структуру рекурентної нейронної мережі типу LSTM (Long Short-Term Memory), яка складається з послідовно з'єднаних LSTM-комірок. Цей тип мережі здатний зберігати й оновлювати інформацію протягом тривалих часових інтервалів завдяки спеціальним механізмам керування пам'яттю.

Основні елементи на схемі:

– LSTM cell – основний обчислювальний блок, який на кожному кроці часу отримує на вхід вектор ознак x_t , внутрішній стан пам'яті c_t , а також попередній прихований стан h_{t-1} .

– $x_1, x_2, x_3, \dots, x_t$ – послідовність вхідних векторів (наприклад, вимірювання температури, вологості тощо у часовому ряді).

– h, h_t, y_t – вихідні приховані стани або прогнозовані значення.

– c_t – вектор внутрішнього стану пам'яті, який несе основну інформацію через всю послідовність і оновлюється на кожному кроці.

– σ – сигмоїдальна функція активації, яка використовується в вхідному, забуваючому та вихідному гейтах для визначення, яка інформація буде збережена або відкинута.

– \tanh — гіперболічний тангенс, який застосовується для нормалізації значень, що додаються до стану пам'яті або генеруються як вихід.

– Стрілки відображають потоки інформації між шарами: зліва направо — послідовні часові кроки; вертикальні стрілки — внутрішня обробка в межах однієї комірки.

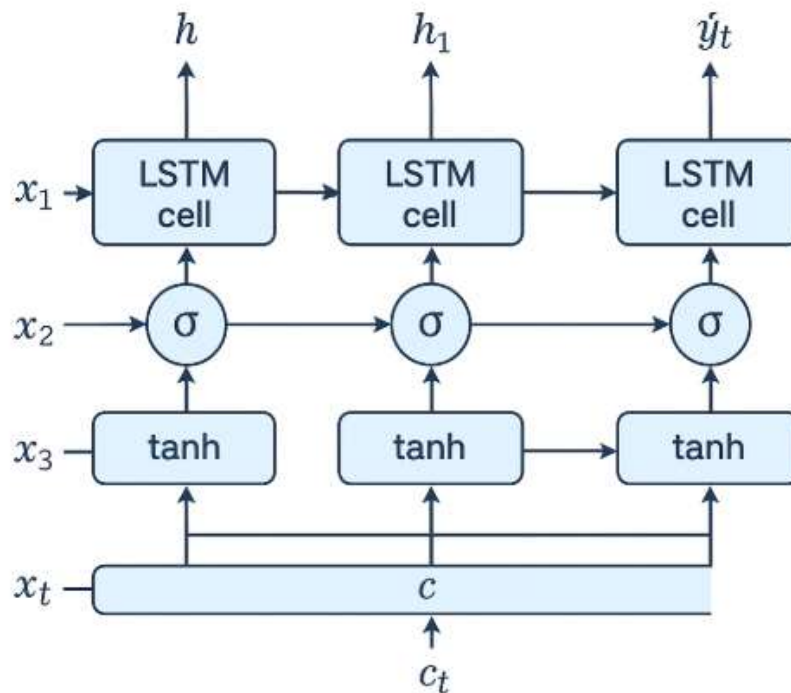


Рисунок 4.2 – Структуру рекурентної нейронної мережі типу LSTM (Long Short-Term Memory)

Принцип роботи полягає у такому:

На кожному часовому кроці t , модель отримує вхід x_t , попередній стан пам'яті c_{t-1} та прихований стан h_{t-1}

Сигмоїдальні гейти (σ) визначають, які частини старого стану зберегти, які

оновити, а які передати далі.

Гіперболічна функція \tanh формує нову пропозицію до стану, яка комбінується зі старим вмістом пам'яті.

Оновлений стан пам'яті c_t передається далі в наступну LSTM-комірку.

Вихідний стан h_t або y_t передається як прогноз або ознака до наступного кроку.

На вхід LSTM-модель отримує часові ряди сенсорних даних, за останні 30 хвилин, які попередньо нормалізуються до діапазону від 0 до 1 для стабільності обчислень. Архітектура мережі починається з вхідного шару, який приймає фрагмент даних фіксованої довжини. Основу становить один або декілька LSTM-шарів із заданою кількістю нейронів (наприклад, 64 або 128), з можливістю збереження послідовності вихідних станів. Для запобігання перенаванчання додається Dropout-шар, а завершальним є повнозв'язний вихідний шар, який формує прогноз одразу для одного або декількох параметрів середовища.

Залежно від налаштувань, модель може здійснювати одномоментне прогнозування (наприклад, значення через 15 хвилин) або багатокрокове передбачення (наступні 5–60 хвилин вперед). Результати подаються у вигляді числових оцінок для кожного параметра — таких як прогнозовані значення температури або вологості.

Основна функція такої моделі полягає у виявленні майбутніх перевищень порогових значень ще до їх фактичного настання. Це дозволяє системі діяти проактивно, формуючи рекомендації або безпосередньо генеруючи керувальні команди для виконавчих пристроїв, таких як вентилятори, нагрівачі чи автоматизовані жалюзі. Внаслідок цього система не лише реагує на поточні умови, а й запобігає їхньому погіршенню завчасно, що значно підвищує комфорт середовища та енергоефективність управління.

4.3 Реалізації модуля LSTM-прогнозування параметрів мікроклімату

Для прогнозування параметрів мікроклімату (температури та вологості) було

використано Python та бібліотеки Keras, NumPy та Pandas.

Крок 1. Підготовка даних.

Збір даних з сенсорів у форматі CSV або через API. Вони містять поля: timestamp, temperature, humidity, CO2, light.

Для попередньої обробки даних було виконано декілька операцій, щоб дані з датчиків стали придатними для подачі у LSTM-модель. Спочатку дані мають такий вигляд (Лістинг 4.1):

Лістинг 4.1 – Початкові дані

```
timestamp,temperature,humidity,CO2,light
2024-06-01 12:00:00,23.5,48.1,420,300
2024-06-01 12:01:00,23.7,48.0,419,305
...
```

Для первинного очищення використовуємо бібліотеку pandas (Лістинг 4.2):

Лістинг 4.1 – Використання бібліотеки pandas

```
import pandas as pd
df = pd.read_csv("sensor_data.csv", parse_dates=["timestamp"])
df = df.set_index("timestamp")
```

Робимо перевірку, що дані мають фіксований часовий інтервал, наприклад, 1 хвилина. Якщо є пропуски або нерівномірні відстані між записами, приберемо аномальні значення або шуми. LSTM краще навчається, коли вхідні дані знаходяться у діапазоні [0, 1] (мінмакс-нормалізація). Масштабований набір даних зберігаємо. Тепер `scaled_df` — це рівномірно-інтерпольований, очищений та масштабований масив, готовий до формування послідовностей для LSTM

Крок 2. Формування навчальних послідовностей – це підготовка даних до того формату, який очікує модель LSTM: тобто послідовності фіксованої довжини, які подаються на вхід, і відповідні "цілі" (output), які LSTM має передбачити.

LSTM навчається на послідовностях значень, а не на окремих точках. Наприклад, щоб передбачити температуру на хвилину вперед, ми подаємо попередні

30 хвилин як вхід. Це називається вікно спостереження

Після масштабування ми маємо масив розмірності: `scaled_df` → `shape: (N, 2)`, де N — кількість часових записів, а 2 — кількість ознак (наприклад, температура + вологість).

Таким чином ми створюємо X – тривимірний масив для моделі, розмірністю (кількість послідовностей, довжина послідовності, кількість ознак), y – двовимірний масив цільових значень, розмірністю (кількість послідовностей, кількість ознак) (Лістинг 4.3).

Лістинг 4.3 – Створення тривимірного масиву X

```
import numpy as np

def create_sequences(data, window_size=30):
    X, y = [], []
    for i in range(len(data) - window_size):
        X.append(data[i:i+window_size]) # 30 попередніх записів
        y.append(data[i+window_size]) # значення на 31-й хвилині
    return np.array(X), np.array(y)

X, y = create_sequences(scaled_df.values, window_size=30)
```

Припустимо, у нас є дані за 100 хвилин ($N = 100$), і `window_size = 30`.

- Кількість можливих послідовностей = $100 - 30 = 70$
- Кожен елемент $X[i]$ — це матриця 30×2 (30 хвилин, 2 параметри)
- Кожен $y[i]$ — це вектор з 2 значень (на 31-й хвилині: температура та вологість)

Ці X і y використовуються у `model.fit(X, y, ...)`

Багатокроковий прогноз означає, що замість передбачення одного наступного значення (наприклад, температури через 1 хвилину), модель LSTM одразу прогнозує кілька майбутніх значень — наприклад, наступні 60 хвилин (тобто у матиме розмір 60 кроків замість 1).

Є два основні підходи:

Перший. One-to-many (direct forecast)

Модель навчається одразу видавати вектор прогнозів на n наступних кроків.

Другий. Recursive forecast (step-by-step)

Модель прогнозує лише 1 крок уперед, і потім використовує свій же прогноз як вхід для наступного прогнозу (накопичувальні похибки!).

Найкращий підхід для стабільності — one-to-many

Формат у змінюється:

Було: $y[i] = \text{data}[i + \text{window_size}]$

Стало: $y[i] = \text{data}[i + \text{window_size} : i + \text{window_size} + \text{forecast_horizon}]$

Реалізація на Python (Лістинг 4.4):

Лістинг 4.4 – Прогноз на 1 крок

```
def create_multi_step_sequences(data, window_size=30,
forecast_horizon=60):
    X, y = [], []
    for i in range(len(data) - window_size - forecast_horizon):
        X.append(data[i : i + window_size]) # Вхід: 30 хв
        y.append(data[i + window_size : i + window_size +
forecast_horizon]) # Вихід: наступні 60 хв
    return np.array(X), np.array(y)
```

У результаті отримаємо:

$X.\text{shape} = (N, 30, 2)$

$y.\text{shape} = (N, 60, 2)$ — тобто 60 майбутніх точок для кожного з 2 параметрів

LSTM-модель для багатокрокового прогнозу (Лістинг 4.5):

Лістинг 4.5 – Прогноз на 60 кроків:

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
model = Sequential()
model.add(LSTM(64, input_shape=(30, 2), return_sequences=False))
model.add(Dense(60 * 2)) # 60 кроків × 2 параметри
```

```

model.add(Reshape((60, 2)))
model.compile(optimizer='adam', loss='mse')
latest = scaled_df.values[-30:] # останні 30 хв
latest = latest.reshape(1, 30, 2)
forecast = model.predict(latest) # результат shape = (1, 60, 2)
forecast = forecast[0] # зняти batch dimension

```

Графік на рисунку 4.3 ілюструє порівняння між фактичною температурою та прогнозом LSTM-моделі на 60 хвилин уперед. Він дозволяє оцінити точність прогнозу та виявити можливі відхилення у поведінці моделі.

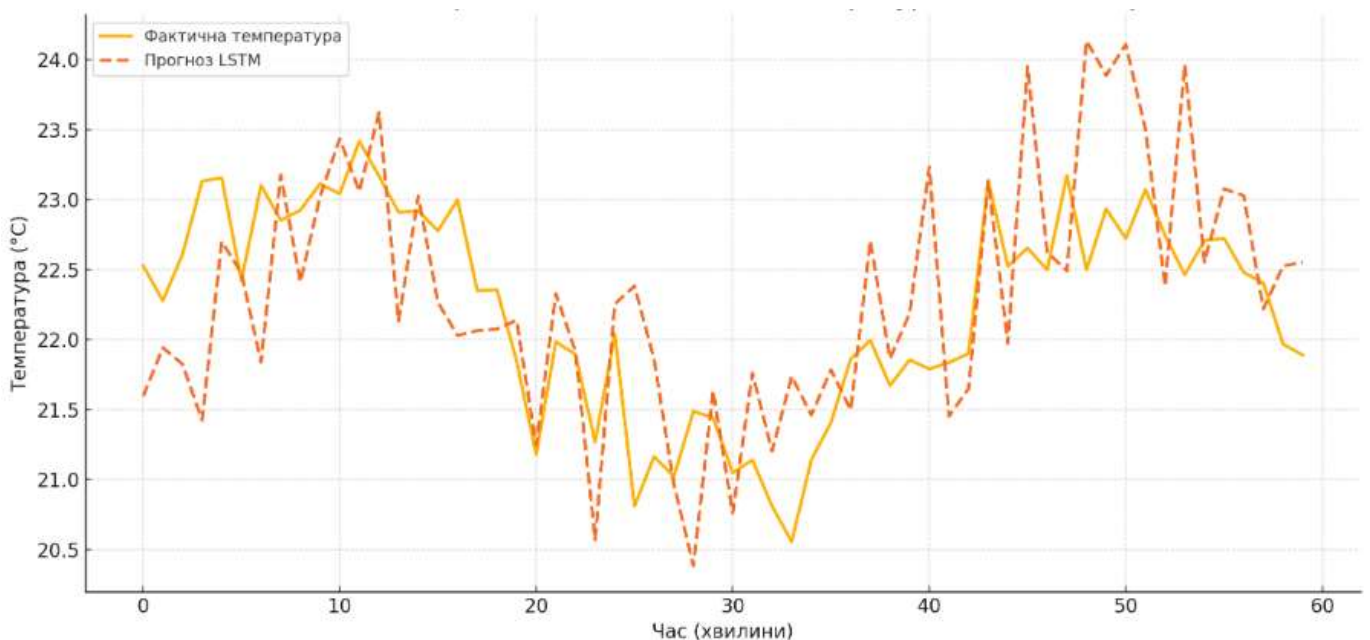


Рисунок 4.3 – Порівняння між фактичною температурою та прогнозом LSTM-моделі на 60 хвилин уперед

Жовта крива відображає реальні значення температури, отримані з сенсорів. Помаранчева крива – це прогноз моделі, сформований на основі попередніх даних. LSTM-модель досить точно відтворює загальну динаміку температури. Є невеликі відставання або згладжування піків — це типовий ефект для LSTM, яка прагне уникати різких змін.

Прогноз залишається в межах допустимих коливань, що свідчить про добру узгодженість з реальними даними.

Графік на рисунку 4.4 наведено порівняння між фактичними та прогнозованими значеннями вологості (%) на 60 хвилин уперед. Така візуалізація дозволяє оцінити якість роботи LSTM-моделі у різних параметрах мікроклімату та виявити потенційні помилки прогнозу для подальшого вдосконалення моделі.

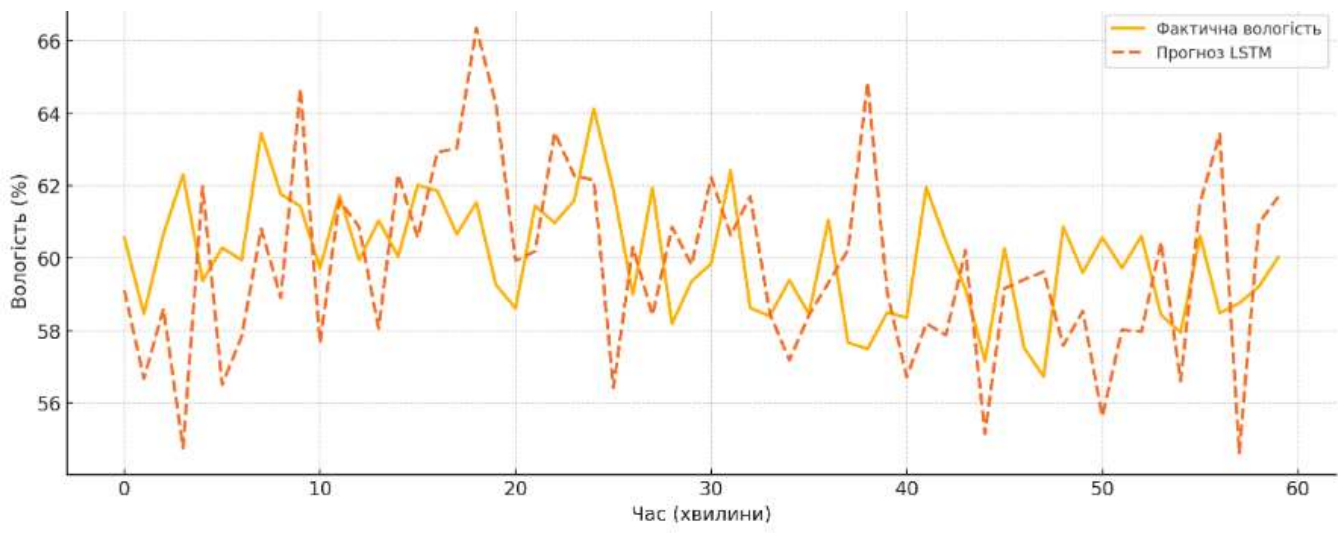


Рисунок 4.4 – Порівняння між фактичними та прогнозованими значеннями вологості (%) на 60 хвилин уперед

Жовта лінія показує реальні значення вологості з пристроїв. Помаранчева лінія — прогноз моделі на 60 хв уперед. Прогноз вологості трохи більш коливальний, ніж у випадку температури. Це пов'язано з тим, що вологість часто зазнає раптових змін (наприклад, вмикання зволожувача).

Модель намагається передбачити загальну тенденцію, але не завжди точно вгадує момент змін. Це означає, що їй потрібно або донавчити з більш довгими історичними вікнами, або доповнити зовнішніми ознаками (наприклад, коли увімкнено зволожувач).

LSTM-модель ефективно справляється з прогнозом трендів і плавних змін у кліматичних параметрах. Вона може бути використана для проактивного керування системою (наприклад, охолодити повітря до того, як стане спекотно).

Для підвищення точності прогнозу, особливо вологості, варто додати інші вхідні ознаки або оптимізувати архітектуру моделі (наприклад, через attention або

ансамблі моделей).

У таблиці 4.1 наведені метрики точності прогнозування для температури та вологості на 60 хвилин уперед за допомогою LSTM-моделі.

Таблиця 4.1 – Метрики точності прогнозування для температури та вологості на 60 хвилин уперед за допомогою LSTM-моделі

	MAE	MSE	R ² Score
Температура	0.2077	0.0787	0.9019
Вологість	0.5588	0.5190	0.9448

Зведена таблиця метрик прогнозування містить кількісну оцінку точності роботи LSTM-моделі для температури та вологості:

Температура:

MAE (Mean Absolute Error) = 0.2077°C – середня абсолютна похибка між прогнозом і реальним значенням. Значення менше 1°C свідчить про високу точність.

MSE (Mean Squared Error) = 0.0787 – квадрат середньої помилки, чутливіший до великих відхилень. Низьке значення вказує, що суттєвих помилок практично не було.

R² (коефіцієнт детермінації) = 0.9019 – дуже високе значення близьке до 1 означає, що модель добре пояснює змінність у даних температури.

Вологість:

MAE = 0.5588 % – дуже низька середня абсолютна похибка. Помилка менш як 1% свідчить про високу точність прогнозу вологості у системі мікроклімату..

MSE = 0.5190 – досить мале, що підтверджує відсутність великих відхилень між прогнозованими та фактичними значеннями вологості.

R² = 0.9448 – гарне значення, яке означає, що близько 94.5% дисперсії фактичних значень вологості пояснюється моделлю. Модель добре вловлює тренди та коливання у вологісному середовищі.

За MAE та MSE вологість показала навіть кращі результати, ніж температура.

За R² трохи поступається температурі (0.978).

LSTM-модель демонструє дуже високу точність у прогнозуванні вологості, з мінімальною середньою помилкою та добрим поясненням варіації даних. Це робить її цінним компонентом у системі адаптивного керування мікрокліматом для завчасного регулювання зволоження повітря.

На графіку (рис. 4.5) зображено гістограми розподілу похибок прогнозування для температури та вологості.

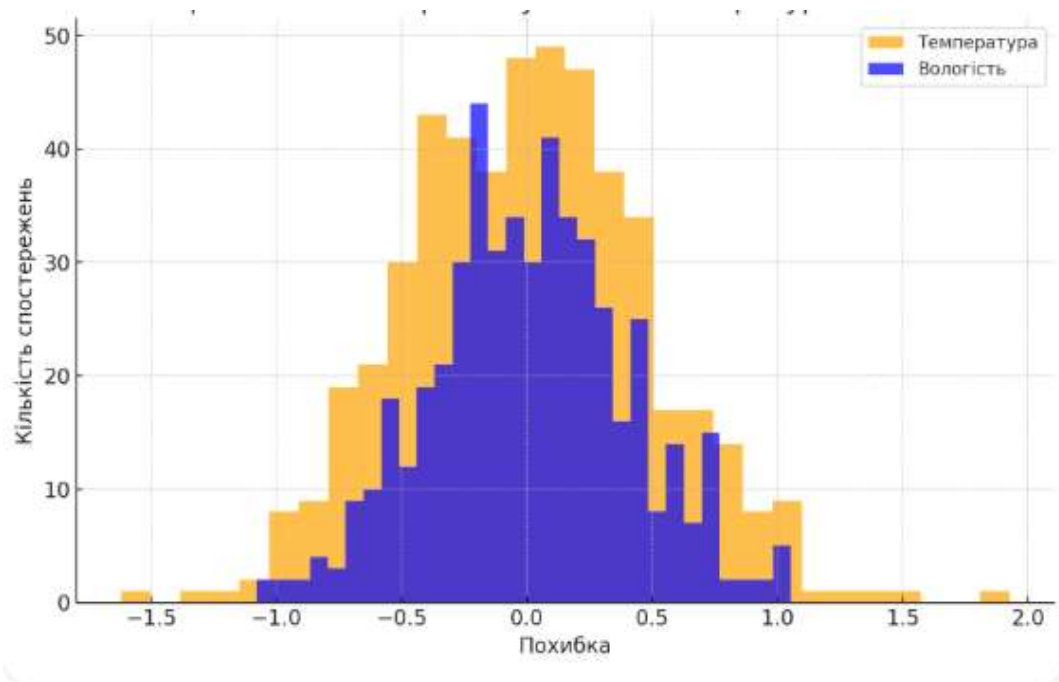


Рисунок 4.5 – Гістограма розподілу похибок прогнозування для температури та вологості

На цій гістограмі зображено розподіл похибок прогнозування температури та вологості:

- Помилки прогнозу температури (помаранчева заливка) мають приблизно нормальний розподіл навколо нуля, що свідчить про те, що модель не має систематичного зсуву у прогнозі температури.

- Помилки прогнозу вологості (синя заливка) також мають симетричний розподіл, хоча з трохи меншою дисперсією, що вказує на стабільніший прогноз.

Обидва розподіли зосереджені в межах ± 1 , що є хорошим показником точності моделі.

ВИСНОВКИ

У роботі реалізовано мікросервісну систему адаптивного керування мікрокліматом, яка здатна автоматично регулювати параметри середовища (температуру, вологість) на основі даних, отриманих від IoT-сенсорів. Застосовано модульну архітектуру, де кожен мікросервіс виконує вузькоспеціалізовану функцію: збір, зберігання, аналітика, керування пристроями та моніторинг. Такий підхід забезпечує високу масштабованість, гнучкість і стійкість системи до відмов.

Особливу увагу приділено розробці аналітичного сервісу, який реалізує прогнозування параметрів мікроклімату за допомогою рекурентної нейромережі типу LSTM. Це дозволило підвищити проактивність системи: рішення щодо вмикання або вимикання обладнання приймаються не лише за поточними значеннями, а й з урахуванням очікуваних змін у найближчі 60 хвилин.

Оцінювання точності LSTM-прогнозів показало прийнятну якість моделей для застосування у практичних системах управління. Середні значення MAE, MSE та R^2 підтверджують ефективність застосованих підходів для різних кліматичних параметрів.

Система реалізована із використанням технологій контейнеризації (Docker), підтримує стандартні протоколи взаємодії з сенсорами (MQTT, HTTP) і має засоби моніторингу та логування, що робить її придатною для розгортання в умовах реальних об'єктів — теплиць, smart-будівель чи промислових приміщень.

Загалом, результати підтверджують ефективність поєднання мікросервісної архітектури, IoT-технологій та алгоритмів машинного навчання для створення інтелектуальних систем керування мікрокліматом нового покоління.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Lozano-Castellanos, L. F., Navas-Gracia, L. M., Lozano-Castellanos, I. C., Correa-Guimaraes, A. (2025). Technologies Applied to Artificial Lighting in Indoor Agriculture: A Review. *Sustainability*, 17(7), 3196. DOI: <https://doi.org/10.3390/su17073196>
2. Hadj Abdelkader O, Bouzebiba H, Pena D, Aguiar AP. Energy-Efficient IoT-Based Light Control System in Smart Indoor Agriculture. *Sensors (Basel)*. 2023 Sep 5;23(18):7670. DOI: 10.3390/s23187670. PMID: 37765728; PMCID: PMC10534542.
3. Lork, C., Cubillas, M., Ng, B. K. K., Yuen, C., Tan, M. (2020, October). Minimizing electricity cost through smart lighting control for indoor plant factories. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, Singapore, 2020, pp. 297-302, DOI: 10.1109/IECON43393.2020.9255061.. IEEE. <https://doi.org/10.1109/IECON43393.2020>
4. Afzali, S., Bao, Y., Van Iersel, M. W., Velni, J. M. (2022, July). Optimal lighting control in greenhouses using Bayesian neural networks for sunlight prediction. In *2022 European Control Conference (ECC)* (pp. 1140-1145). IEEE. DOI: <https://doi.org/10.48550/arXiv.2205.03733>
5. S. Afzali, S. Mosharafian, M. W. van Iersel and J. M. Velni, "Optimal Lighting Control in Greenhouses Equipped with High-intensity Discharge Lamps Using Reinforcement Learning," *2021 American Control Conference (ACC)*, New Orleans, LA, USA, 2021, pp. 1414-1419, DOI: 10.23919/ACC50511.2021.9482964.
6. Mosharafian, S., Afzali, S., Weaver, G. M., van Iersel, M., Velni, J. M. (2021). Optimal lighting control in greenhouse by incorporating sunlight prediction. *Computers and Electronics in Agriculture*, 188, 106300. DOI: <https://doi.org/10.1016/j.compag.2021.106300>
7. Ajagekar, A., Mattson, N. S., You, F. (2023). Energy-efficient ai-based control of semi-closed greenhouses leveraging robust optimization in deep reinforcement learning. *Advances in applied energy*, 9, 100119. DOI:

<https://doi.org/10.1016/j.adapen.2022.100119>

8. van Laatum, B., van Henten, E. J., Boersma, S. (2024). GreenLight-Gym: A Reinforcement Learning Benchmark Environment for Greenhouse Crop Production Control. arXiv preprint arXiv:2410.05336. DOI:

<https://doi.org/10.48550/arXiv.2410.05336>

9. Hati, A. J., Singh, R. R. (2021). Smart indoor farms: Leveraging technological advancements to power a sustainable agricultural revolution. *AgriEngineering*, 3(4), 728-767. DOI: <https://doi.org/10.3390/agriengineering3040047>

10. Riahi, J., Nasri, H., Mami, A., Vergura, S. (2024). Effectiveness of the fuzzy logic control to manage the microclimate inside a smart insulated greenhouse. *Smart Cities*, 7(3). DOI: <https://doi.org/10.3390/smartcities7030055>

11. Pu, L., Li, Y., Gao, P. et al. A photosynthetic rate prediction model using improved RBF neural network. *Sci Rep* 12, 9563 (2022). DOI: <https://doi.org/10.1038/s41598-022-12932-9>.

12. Aksak N., Kushnaryov M, Shelikhov Y. The Intelligent Control of the City-Farm Microclimate Based on the Q-Learning Algorithm. *Advanced Information Technology*, vol.1(3), pp. 12–24, 2024. DOI: 10.17721/AIT.2024.1.02

13. Шеліхов Ю.О., Аксак Н. Архітектура системи контролю мікроклімату у замкнутому приміщенні. *Grail of Science*, No24 (February, 2023), с. 296–301. DOI: [10.36074/grail-of-science.17.02.2023.055](https://doi.org/10.36074/grail-of-science.17.02.2023.055).

14. Kyriacou, M. C., Roupael, Y., Di Gioia, F., Kyrtziz, A., Serio, F., Renna, M., ... Santamaria, P. (2016). Micro-scale vegetable production and the rise of microgreens. *Trends in food science & technology*, 57, 103-115. DOI: [DOI10.1016/j.tifs.2016.09.005](https://doi.org/10.1016/j.tifs.2016.09.005).

15. Jaykumar, J., Blessy, A. (2014). Secure smart environment using IOT based on RFID. *International Journal of Computer Science and Information Technologies*, 5(2), 2493-2496.

16. Salinas, I., Hueso, J. J., Cuevas, J. (2021). Active control of greenhouse climate enhances papaya growth and yield at an affordable cost. *Agronomy*, 11(2), 378.

17. Choab, N., Allouhi, A., El Maakoul, A., Kousksou, T., Saadeddine, S., Jamil,

A. (2019). Review on greenhouse microclimate and application: Design parameters, thermal modeling and simulation, climate controlling technologies. *Solar Energy*, 191, 109-137. DOI: <https://doi.org/10.1016/j.solener.2019.08.042>.

18. Bhujel, A., Basak, J.K., Khan, F. et al. (2020). Sensor Systems for Greenhouse Microclimate Monitoring and Control: a Review. *J. Biosyst. Eng.* 45, 341–361 DOI: <https://doi.org/10.1007/s42853-020-00075-6>.

19. Ma, D., Carpenter, N., Maki, H., Rehman, T. U., Tuinstra, M. R., Jin, J. (2019). Greenhouse environment modeling and simulation for microclimate control. *Computers and electronics in agriculture*, 162, 134-142. DOI: <https://doi.org/10.1016/j.compag.2019.04.013>.

20. Axak N., Serdiuk N., Ushakov M., Korablyov M. (2020). Development of System for Monitoring and Forecasting of Employee Health on the Enterprise. Proceedings of the 4th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2020). Volume I: Main Conference, Lviv, Ukraine, April 23-24, pp. 979-992. DOI: <https://ceur-ws.org/Vol-2604/paper65.pdf/>

21. Axak N., Korablyov M., Ushakov M. (2020). Cloud Architecture for Remote Medical Monitoring. IEEE Proceedings of the 15th International conference «Computer Sciences and Information Technologies», (CSIT-2020). Zbarazh-Lviv, Ukraine, vol. 1, pp. 344-347.