

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління  
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)  
(рівень вищої освіти)

Система моніторингу кліматичних показників навколишнього  
середовища  
(тема)

Виконав: студент 2 курсу, групи СКСм-21-1

Надиршин Р.Ю.  
(прізвище, ініціали)

Спеціальність 123 – Комп'ютерна інженерія  
(код і повна назва спеціальності)


Тип програми освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Спеціалізовані  
комп'ютерні системи  
(повна назва освітньої програми)

Керівник проф., д.т.н., Чумаченко С.В.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

  
(підпис)

Чумаченко С.В.  
(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління

Кафедра Автоматизації проектування обчислювальної техніки


Рівень вищої освіти другий (магістерський)

Спеціальність 123 Комп'ютерна інженерія  
(шифр і назва)

Тип програми Освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Спеціалізовані комп'ютерні системи  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри   
(підпис)

« 12 » грудня 20 22 р.

## ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Надиршину Руслану Юрійовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Система моніторингу кліматичних показників  
навколишнього середовища

затверджена наказом по університету від 14 листопада 2022 р. № 1478

2. Термін подання студентом роботи до екзаменаційної комісії 12 грудня 2022 р.

3. Вихідні дані до роботи \_\_\_\_\_

Мікроконтролер ESP8266

Датчик BME280

Апаратно-серверна взаємодія

Клієнт-серверна взаємодія

Віддалений сервер

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

Аналіз предметної області та постановка задачі

Аналіз літератури

Розробка та програмування апаратної частини

Розробка алгоритмів

Розробка веб-додатку

Тестування розробленої системи

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 20 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

7. Дата видачі завдання 01.09.2022

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Видача теми роботи, узгодження і затвердження	01.09.2022 – 04.09.2022	
2	Аналіз літератури	10.09.2022 – 15.10.2022	
3	Розробка алгоритмів	16.10.2022 – 27.10.2022	
4	Програмування апаратної частини	03.10.2022 – 05.10.2022	
5	Розробка веб-додатку	15.11.2022 – 01.12.2022	
6	Проведення тестування отриманої моделі	02.12.2022 – 05.12.2022	
7	Оформлення пояснювальної записки	06.12.2022 - 11.12.2022	
8	Перевірка виконаного проекту керівником	12.12.2022	
9	Захист проекту	16.12.2022	

Студент   
(підпис)

Керівник роботи  проф., д.т.н., Чумаченко С.В.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 59 с., 16 рис., 1 табл., 4 дод., 19 джерел посилання.

МІКРОКОНТРОЛЕР, ARDUINO IDE, NODEMCU, ІОТ, СКЕТЧ, FRONTEND, BACKEND, VPS, API, AJAX

Метою роботи є підвищення якості збирання, збереження й обробки достовірної та оперативної інформації шляхом розробки системи моніторингу кліматичних показників навколишнього середовища, що спрямовано на заходи з попередження та зменшення негативних наслідків змін стану навколишнього середовища.

Система моніторингу на основі мікроконтролера дозволить мінімізувати часові та матеріальні витрати для організації виконуваних задач, а також створювати інноваційні науково-технічні рішення соціальних та екологічних проблем світової спільноти.

## ABSTRACT

Master's thesis contains: 59 pages, 16 figures, 1 tables, 4 appendices, 19 sources of reference.

МИКРОКОНТРОЛЕР, ARDUINO IDE, NODEMCU, IOT, SKETCH, FRONTEND, BACKEND, VPS, API, AJAX

The purpose of the work is to improve the quality of collection, preservation and processing of reliable and operational information by developing a system for monitoring climatic indicators of the environment, which is aimed at measures to prevent and reduce the negative consequences of changes in the environment.

A monitoring system based on a microcontroller will allow to minimize time and material costs for the organization of performed tasks, as well as to create innovative scientific and technical solutions to social and environmental problems of the world community.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ	
I ТЕРМІНІВ .....	8
ВСТУП.....	9
1 ОГЛЯД ЛІТЕРАТУРИ.....	11
1.1 Моніторинг клімату і його завдання.....	11
1.2 Становлення і розвиток моніторингу довкілля.....	12
1.3 Огляд існуючих рішень .....	18
1.4 Типи систем моніторингу навколишнього середовища.....	19
1.5 IoT системи моніторингу навколишнього середовища.....	20
2 ОПИС ОБРАНОЇ СИСТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ.....	26
2.1 Постанова задачі .....	26
2.2 Опис обраної системи.....	27
3 РЕАЛІЗАЦІЯ ПРОЕКТУ .....	31
3.1 Розробка алгоритмів .....	31
3.1.1 Алгоритми апаратно-серверної взаємодії .....	31
3.1.2 Алгоритми клієнт-серверної взаємодії.....	34
3.2 Схема пристрою .....	34
3.3 Реалізація методів .....	36
3.3.1 Скетч Arduino IDE для плати NodeMCU.....	36
3.3.2 Файлова структура проекту.....	39
3.3.3 Реалізація бекенд коду веб-додатку .....	41
3.3.4 Реалізація фронтенд коду веб-додатку.....	43
4 ТЕСТУВАННЯ.....	45
4.1 Постанова задач на тестування.....	45
4.2 Тестування процесу вимірювання показань в різних умовах .....	46
4.2.1 Тестування в статичних кімнатних умовах .....	46
4.2.2 Тестування в умовах низької температури .....	47

4.2.3 Тестування в умовах високої температури .....	48
4.2.4 Тестування з різним значенням інтервалу збереження даних .....	49
4.3 Тестування функціоналу веб-сторінки .....	50
4.3.1 Тестування відображення погоди різних міст .....	51
4.3.2 Тестування параметрів сортування .....	51
4.3.3 Тестування графіків .....	53
4.4 Результати тестування .....	54
ВИСНОВКИ .....	55
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	57
Додаток А Графічний матеріал кваліфікаційної роботи.....	60
Додаток Б Скріншоти веб-інтерфейсу .....	70
Додаток В Код програми.....	72
В.1 Файл script.js .....	72
В.2 Файл request.js.....	77

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

IoT – Інтернет речей (англ. Internet of Things)

ESP8266 – мікроконтролер з інтерфейсом Wi-Fi та 80 МГц 32-бітним процесором

Скетч – програма, яка написана у середовищі Arduino IDE

гПа (гектопаскаль) –  $10^2$  Па (паскаль) – одиниця виміру тиску

Frontend – стадія веб-розробки, яка відповідальна за роботу з боку користувача (стиль, інтерфейс, функціонал тощо)

Backend – стадія веб-розробки, яка відповідальна за роботу з боку сервера (взаємодія з даними сервера, базою даних)

VPS (англ. virtual private server – віртуальний виділений сервер) – це повноцінна альтернатива фізичного виділеного сервера з великою кількістю переваг, високою стабільністю, простотою в управлінні і налаштуванні, стійкістю до відмов і набагато меншими фінансовими витратами

API (англ. Application Programming Interface) – це набір чітко визначених методів для взаємодії різних компонентів.

AJAX (Asynchronous Javascript And Xml) – технологія звернення до сервера без перезавантаження сторінки.

## ВСТУП

Функціонування ефективної системи моніторингу довкілля є невід'ємною складовою державної політики у сфері охорони навколишнього природного середовища. Основною метою проведення моніторингу довкілля є збирання, збереження та обробка достовірної та оперативної інформації, необхідної для розробки заходів із попередження та зменшення негативних наслідків змін стану навколишнього середовища. [1]

Для вирішення цієї проблеми розробляються більш доступні та досконалі системи моніторингу довкілля. Їх перевагою також є можливість встановлення не лише для державних метеорологічних станцій, а і у якості побутових приладів.

Оперативний аналіз даних на основі моніторингу навколишнього середовища дає можливість взаємодіяти з системами безпеки, вентиляції та клімат-контролю системи розумного будинку, а також виконувати прогнозування зміни погоди. Сучасні системи управління кліматом є досить складними системами автоматичного регулювання, що здійснюють підтримання параметрів повітря в контрольованих об'ємах у заданих межах на підставі сигналів, що надходять з датчиків температури та вологості. Розвиток та модернізація систем моніторингу сприятиме підвищенню якості збирання, збереження та обробки достовірної та оперативної інформації, необхідної для розробки заходів із попередження та зменшення негативних наслідків змін стану навколишнього середовища.

Домашні метеостанції порівняно нещодавно з'явилися на ринку. Їх родоначальниками є звичайні барометри. Домашня метеостанція обробляє дані, що надходять з одного або декількох датчиків, які можуть бути розташовані в приміщенні або поза ним. Вони вимірюють температуру, атмосферний тиск, вологість повітря. Деякі домашні метеостанції мають

можливість формувати прогноз погоди на добу, обробляючи дані з датчиків.

Персональні метеостанції зазвичай мають цифрову консоль, яка забезпечує зчитування даних, що збираються. Ці консолі можуть взаємодіяти з персональним комп'ютером, де дані можна відображати, зберігати та завантажувати на веб-сайти або в системи передачі/розподілу даних.

Метою даної роботи є підвищення якості збирання, збереження, обробки достовірної та оперативної інформації шляхом аналізу існуючих та розробки власної системи моніторингу кліматичних показників навколишнього середовища, що спрямовано на заходи з попередження та зменшення негативних наслідків для навколишнього середовища.

## 1 ОГЛЯД ЛІТЕРАТУРИ

### 1.1 Моніторинг клімату і його завдання

При глобальних перетвореннях довкілля визначальну роль відіграють кліматичні зміни, спричинені природними та антропогенними факторами (парниковий ефект, озонові діри, техногенні забруднення довкілля). Перебуваючи в тісному взаємозв'язку з усіма компонентами природного середовища, клімат (багаторічний режим погоди, властивий даній місцевості) відчутно впливає на них, на умови життя і самопочуття людини.

З'ясування антропогенних змін і коливань клімату неможливе без вивчення його природної динаміки, яка ґрунтується на даних про стан кліматичної системи «атмосфера – океан – поверхня суші (з річками й озерами) – літосфера біота» і взаємодію елементів цієї системи за тривалий час. Спостереження за станом кліматичної системи, оцінювання та прогнозування її подальшого розвитку здійснюють за допомогою кліматичного моніторингу.

Кліматичний моніторинг пов'язаний з екологічним. Він потребує спеціальної системи спостережень, спроможної забезпечити виконання наукових і практичних завдань, зокрема надати широку кліматичну інформацію. З цією метою, як правило, створюють службу збору кліматичних даних, сфера діяльності якої простягається і за межі моніторингу антропогенних змін клімату. Для пізнання сутності й антропогенної складової змін і коливань клімату необхідний великий масив даних про параметри елементів біосфери, процеси, які характеризують її зміни. Особливо це важливо при простежуванні змінюваності клімату в просторі і часі. Прогнозування сезонних і річних коливань клімату відбувається на основі інформації яку забезпечує глобальна система

спостережень. При цьому слід мати на увазі, що спостереження, спрямовані на вивчення змінюваності клімату, повинні обов'язково враховувати інерційність кліматичної системи [2].

Моніторинг клімату зосереджується на реалізації таких завдань:

- збирання даних про стан кліматичної системи;
- аналізування і оцінювання природних та антропогенних змін і коливань клімату (включаючи порівняння клімату минулого з сучасним);
- виокремлення антропогенних ефектів у зафіксованих змінах клімату;
- виявлення природних і антропогенних факторів, що зумовлюють зміну клімату;
- виявлення критичних елементів біосфери, вплив на які може спричинити кліматичні зміни.

## 1.2 Становлення і розвиток моніторингу довкілля

Моніторинг довкілля виник у другій половині ХХ ст. як науково-практичний напрям системної екології, завданням якої є встановлення критеріїв і виявлення меж стійкості екологічних систем. Тоді його метою було отримання репрезентативних даних про стан, динамічні зміни екосистем, створення бази даних, вибір об'єктів і формування мережі спостережень. На той час поняття “моніторинг довкілля” охоплювало не лише систему постійних спостережень за станом компонентів біосфери, а й засновану на природничо-науковій основі (біологічній, фізико-хімічній, геофізичній) певну їх методологію, а також означало дієвий засіб охорони довкілля.

На початку 70-х років ХХ ст. було обґрунтовано альтернативні концепції моніторингу довкілля як сфери наукового знання і практичної діяльності.

Згідно з концепцією сучасного російського геофізика Ю. Ізраеля

моніторинг довкілля є системою цілеспрямованих, періодично повторюваних і програмованих спостережень за одним і більше елементами навколишнього середовища у просторі і часі. Основними елементами цієї системи є спостереження, оцінювання і прогнозування стану довкілля. Моніторинг формується з певних підсистем, серед яких особливу роль відіграє екологічний моніторинг. За цією концепцією метою моніторингу є фіксація антропогенних змін природного середовища, а управління його якістю не передбачено.

За переконаннями російського географа-грунтознавця І. Герасимова, моніторинг довкілля – це організована на різних рівнях система спостережень, контролювання і управління його станом. Налагодження моніторингу довкілля відповідно до цієї концепції сприяє виявленню екологічних небезпек. Відповідно до завдань і масштабів об'єктів спостереження І. Герасимов розрізняє такі рівні моніторингу довкілля:

– біоекологічний (санітарно-гігієнічний) моніторинг. Його сутність полягає у спостереженні за станом і впливом довкілля на здоров'я людини з метою захисту її від негативних чинників;

– геоекологічний (геосистемний, природно-господарський) моніторинг. Передумовами його реалізації є спостереження за змінами природних екосистем і перетворенням їх на природно-технічні, прогнозування стихійних змін навколишнього середовища і явищ, які погіршують життєве середовище людей;

– біосферний моніторинг. Здійснюють його шляхом спостереження за природними процесами і явищами на рівні біосфери, а також через з'ясування глобальних змін фонових показників у природі.

М. Голубець – український вчений, спеціаліст у галузях ботаніки й екології, розглядає моніторинг довкілля як багаторівневу систему спостереження, оцінювання і прогнозування стану навколишнього природного середовища, розроблення науково обґрунтованих рекомендацій

для прийняття ефективних природоохоронних управлінських рішень. Він поділяє моніторинг на ієрархічні рівні. Найвищий рівень – класифікація моніторингу за територіально-просторовими параметрами контрольованих процесів, тобто масштабами спостережень. За цим критерієм вчений виділяє глобальний, материковий, океанічний, міжнародний, національний, регіональний, локальний типи моніторингу довкілля.

Дослідження реакцій абіотичної (неживої матерії) і біотичної (живих організмів) складових біосфери на дію природних і антропогенних факторів М. Голубець об'єднує у групи моніторингу довкілля, які він розрізняє залежно від складових довкілля (біологічна, гідрометеорологічна, біоенергетична, біогеохімічна тощо). До найнижчого рівня належать види моніторингу довкілля (ботанічний, зоологічний, гідробіологічний, мікробіологічний), сутність яких полягає у спостереженні за реакцією організмів на зміни, що відбуваються у біосфері.

У період з 1972 по 1974 рік науковий комітет з проблем навколишнього середовища Міжнародної ради наукових союзів (SCOPE) виробив і запропонував ідею глобального моніторингу. Теоретично обґрунтував її американський вчений Р. Мунн, який розглядав моніторинг як систему контролю за навколишнім середовищем, що охоплює спостереження за його станом, визначення можливих змін і розроблення заходів з управління довкіллям. Сутність концепції глобального моніторингу полягає в необхідності здійснення повторних спостережень за елементами навколишнього середовища в просторі і часі з певною метою за конкретними програмами. На основі цієї концепції виникли різноманітні підсистеми моніторингу довкілля: моніторинг приземного й верхнього шарів атмосфери; атмосферних опадів; гідросфери (поверхневих вод суші, вод океанів, морів і підземних вод); літосфери (передусім ґрунту); озонового шару; океану; кліматичний моніторинг; геофізичний; фізичний; біогеохімічний.

У 1986 р. Секретаріат ООН з навколишнього середовища,

послугуючись розробками Р. Мунна, видав “Довідник з екологічного моніторингу”, який містить методики і програми моніторингу для країн, що розвиваються. Запропоновані у Довіднику системи моніторингу ґрунтуються на природничо-наукових дослідженнях і передбачають:

– виявлення і дослідження природних ресурсів, які забезпечують виробництво продуктів харчування (моніторинг клімату, рельєфу, ґрунтів, рослинності, популяцій);

– дослідження природних умов (моніторинг ерозії ґрунтів, твердого стоку).

Розвинуті країни запровадили моніторинг довкілля у 60–70-ті роки ХХ ст. В усіх розвинутих країнах моніторинг довкілля здійснюється на основі рекомендацій ООН з урахуванням національних особливостей.

В Україні розроблення і впровадження системи екологічного моніторингу розпочалося у 1992 р. відповідно до закону “Про охорону навколишнього природного середовища” та “Положення про державний моніторинг навколишнього середовища”. Ця система передбачає спостереження за довкіллям, збирання, обробку і оцінювання отриманих даних та прогнозування його стану, формування відповідних баз інформації, розробку на їх основі науково обґрунтованих природоохоронних заходів, передбачення надзвичайних ситуацій техногенного і природного характеру, створення безпечних умов життєдіяльності людини.

Отже, моніторинг довкілля є дієвим засобом природоохоронної політики, яка здійснюється відповідно до екологічних прогнозів [3].

### 1.3 Види кліматичного моніторингу

Кліматичний моніторинг охоплює геофізичний (система спостережень за абіотичною частиною біосфери: кліматом, рельєфом, температурою,

сонячною радіацією тощо) та біологічний (система спостережень за станом біотичної складової біосфери та її реакцією на антропогенний вплив) моніторинги. У його здійсненні важливу роль відіграють метеорологічні служби, які сформовані з наземних та супутникових підсистем.

Усі основні кліматичні дані, необхідні для аналізу змін клімату, поділяють на такі групи: вимірювання основних метеорологічних параметрів, вивчення та аналіз атмосферних явищ і процесів, які характеризують відповідний стан погоди; моніторинг стану кліматичної системи; моніторинг внутрішніх та зовнішніх факторів, які впливають на клімат і стан кліматичної системи; моніторинг можливих фізичних і екологічних перетворень у довкіллі, які відбуваються внаслідок кліматичних змін і коливань.

Вимірювання основних метеорологічних параметрів, вивчення та аналіз атмосферних явищ і процесів, які характеризують відповідний стан погоди. До цієї групи належать дані про температуру і вологість повітря, атмосферний тиск, швидкість та напрямок вітру, інтенсивність опадів, гідрологічні дані, а також дані про сніговий покрив, вологість і глибину промерзання ґрунту та інші, які отримують на метеорологічних і гідрологічних станціях і постах. На основі цієї інформації здійснюють моніторинг атмосферних явищ і процесів.

На сучасному етапі у світі функціонує 40 000 кліматологічних і 140 000 дощомірних станцій, однак розміщені вони нерівномірно. Міжнародний обмін основними погодними даними забезпечують Всесвітня служба погоди (ВСП), Всесвітня метеорологічна організація (ВМО). Глобальну систему спостережень формують наземна й супутникова підсистеми. Наземна підсистема базується на опорній синоптичній мережі. Вона отримує інформацію також з кораблів і літаків, метеорологічних радіолокаторів, різних систем зондування атмосфери. До наземної підсистеми спостережень належать станції вимірювання сонячної радіації і фонового забруднення

атмосфери. Супутникову підсистему утворюють геостаціонарні і розташовані на навколополярних орбітах метеорологічні супутники. Вони відстежують вертикальні профілі температури і вологості, температуру поверхні моря, поверхні суші та верхнього шару хмар, сніговий покрив, радіаційний баланс.

Моніторинг стану кліматичної системи (реакція кліматичної системи та її елементів на природні й антропогенні зміни). Він охоплює всю біосферу, але зосереджений на реакціях, які безпосередньо стосуються антропогенних змін клімату. Спостереження за станом клімату охоплює моніторинг кліматоутворюючих факторів, а також величин, які характеризують реакцію кліматичної системи та її елементів на різні дії, передусім антропогенні.

Моніторинг внутрішніх та зовнішніх факторів (особливо антропогенних факторів та їх джерел), які впливають на клімат і стан кліматичної системи. До зовнішніх факторів належать вплив Сонця і космічного випромінювання. До внутрішніх факторів, які впливають на клімат і кліматичну систему, відносять теплові викиди та викиди забруднюючих речовин у біосферу, їх перерозподіл між різними середовищами.

Моніторинг можливих фізичних і екологічних перетворень у довкіллі, які відбуваються внаслідок кліматичних змін і коливань. Трансформації клімату, впливаючи на стан біосфери, позначаються і на господарській діяльності людини. Чутливими до коливань клімату є елементи біосфери, розташовані у полярних широтах, засушливих місцях, а також екосистеми пустельних зон, високогір'їв.

Параметри змін у біосфері називають непрямими показниками змін клімату. Цю групу показників утворюють зміни рівня морів, озер, розташування берегової лінії, річних шарів, донних відкладень озер, снігової лінії та ін. До неї також зараховують і такі екологічні ознаки, як зміна характеру рослинності, врожайності культур, морської мікрофлори і мікрофауни, популяцій комах, особливостей поширення хвороб тварин і

рослин, передусім у зонах з найбільшою чутливістю до змін клімату.

Комплекс усіх цих кліматичних даних необхідний для проведення всебічного аналізу стану довкілля і моделювання клімату, на підставі яких виокремлюють критичні фактори впливу і найчутливіші елементи біосфери, що є передумовою оптимального функціонування системи кліматичного моніторингу [4].

### 1.3 Огляд існуючих рішень

Сучасна метеостанція – це сукупність різних датчиків, які вимірюють погодні (метеорологічні) параметри. Метеорологічні станції поділяються на аналогові та цифрові.

Останнім часом в світі спостерігається стрімкий зріст кількості виробників та різних моделей домашніх метеостанцій. Існують сотні моделей домашніх цифрових метеостанцій на будь-який смак, з різними наборами функцій користувача, вартість яких варується від десятків до сотень доларів США. Найбільш просунуті (тому і найбільш дорогі) домашні метеостанції, які крім зазначених параметрів визначають кількість опадів, швидкість і напрям вітру, рівень УФ-радіації, а також дозволяють відправляти отримані дані іншим користувачам (мережею Інтернет або за допомогою аматорського радіозв'язку), і використовують не лише власні датчики температури і вологості, але і отримують дані прогнозу погоди на 3-5 доби за сигналами супутника (технологія WeatherDirect). [5]

Домашні метеостанції порівняно нещодавно з'явилися на ринку. Їх родоначальниками є звичайні барометри. Домашня метеостанція обробляє дані, що надходять з одного або декількох датчиків, які можуть бути розташовані в приміщенні або поза ним. Вони вимірюють температуру, атмосферний тиск, вологість повітря. Деякі домашні метеостанції мають можливість формувати прогноз погоди на добу, обробляючи дані з датчиків.

Персональні метеостанції зазвичай мають цифрову консоль, яка забезпечує зчитування даних. Ці консолі можуть взаємодіяти з персональним комп'ютером, де дані можна відображати, зберігати та завантажувати на веб-сайти або в системи передачі/розподілу даних.

#### 1.4 Типи систем моніторингу навколишнього середовища

Серед існуючих рішень побутові метеостанції діляться на механічні та цифрові, в яких функціонал та точність вимірювань залежить від вартості та складності системи:

а) Механічні аналогові метеостанції використовуються для показань температури, вологості і атмосферного тиску. За їх допомогою можна передбачити зміну погоди та опадів, аналізуючи зміну тиску.

б) Цифрові метеостанції окрім барометра та термометра, без яких прогноз погоди неможливий, додатково мають годинник, календар, будильник. Такі метеостанції діляться на декілька груп з різним ціновим діапазоном:

– бюджетні пристрої мають урізаний функціонал, який базується на наявності сенсорів термометра та барометра. Деякі моделі мають зовнішній провідний датчик. На дисплеї можна побачити температуру, тиск, прогноз погоди на найближчі 12 годин;

– пристрої середнього цінового діапазону володіють більшим функціоналом, у наявності один або декілька безпроводних термогігросенсорів, датчики на яких мають хороший радіус дії. Крім цього, такі пристрої мають додаткові можливості з різними типами календаря, підсвічуванням, проектором;

– професійні станції є найбільш дорогими пристроями кліматичного контролю. Вони мають обширний функціонал за рахунок використання додаткових спеціалізованих датчиків та інструментів. Такі моделі можуть

вимірювати швидкість та напрям вітру, висоту над рівнем моря, що допомагає у більш точному прогнозуванні погоди. Крім цього, вони, як правило, мають додаткову пам'ять та можливість підключення до ПК за допомогою USB-інтерфейсу. Метеостанції Wi-Fi також відносяться до таких пристроїв [6].

Моніторинг середовища є важливою функцією для об'єктів усіх типів, включаючи IT-центри обробки даних, склади, школи, лікарні, офіси, ресторани, ферми, фабрики, зоопарки, музеї тощо. Підтримка стабільного середовища, де розміщено обладнання та/або продукти, має вирішальне значення для їх подальшої роботи, захисту та використання. Зважаючи на це, моніторинг зазвичай потрібен на постійній основі. Ринок «IT-моніторингу середовища та моніторингу об'єктів» є, мабуть, однією з найбільш швидкозростаючих ніш у комп'ютерній індустрії.

У моніторингу IT-середовища та об'єктів існує сім основних загроз, які постійно викликають найбільше занепокоєння. До них належать температура, потужність, вологість, повінь, дим, потік повітря та вхід у приміщення. Для багатьох цілей можна знайти додаткові типи датчиків. Мета полягає в тому, щоб запровадити датчики, які відповідатимуть конкретним потребам програми моніторингу та інтегрувати їх з автоматизованою системою моніторингу та оповіщення. [7]

### 1.5 IoT системи моніторингу навколишнього середовища

Технологія моніторингу погоди на основі Інтернету речей розвивається дуже швидко, в деяких параметрах вона вже обійшла функції звичайних систем. Завдяки можливості використання віддаленого моніторингу та розширеної аналітики ця технологія робить революцію на різних підприємствах, пропонуючи низку переваг.

Інновації в моніторингу погоди зосереджені на контролі та моніторингу

різних погодних умов. Пристрої IoT допомагають вимірювати фізичні параметри певного місця та завантажувати їх у режимі реального часу, внаслідок чого дані можна негайно проаналізувати.

Ці системи дозволяють здійснювати моніторинг клімату в складних географічних рельєфах. Сенсорні пристрої, розміщені в певних місцях, можуть виконувати всю роботу для виявлення поточних деталей клімату, таких як вологість повітря, кількість опадів, вологість ґрунту, швидкість вітру, рівень CO<sub>2</sub> тощо, які необхідні для прогнозування погоди. Ця технологія також може зробити можливим безпечний моніторинг екстремальних погодних умов у негостинному середовищі.

Моніторинг погоди та його прогнозування стали життєво важливою частиною повсякденного життя через його численні застосування в сільському господарстві, рибальстві, судноплавстві та військових операціях. Вимірювання погоди за допомогою звичайних або ручних станцій моніторингу погоди вимагає кваліфікованої робочої сили та регулярного обслуговування, що незмінно збільшує вартість життєвого циклу станції моніторингу погоди. Крім того, на базовій станції отримані дані реєструються та завантажуються на онлайн-сервер даних, щоб забезпечити повсюдний доступ до вимірювань погоди в усьому світі [8].

Персональна метеостанція (англ. PWS – Personal Weather Station) — це відкрита метеостанція, обладнана: термометром, барометром, анемометром, гігрометром і флюгерами. Деякі більш просунуті моделі також будуть включати кремнієвий фотодіодний піранометр, ультрафіолетовий індекс і/або датчики якості повітря. PWS належать приватним особам (наприклад, ентузіастам погоди) або організаціям, які добровільно передають свої дані. Дані PWS регулярно завантажуються та доступні на таких веб-сайтах, як WeatherLink, Netatmo Weathermap, Weather Underground або EUweather. Оскільки більшість із цих пристроїв повідомляють кожні 5-15 хвилин, дані є загальнодоступними безкоштовно швидше, ніж інші відкриті джерела даних

про сонячну радіацію, тобто супутникові або офіційні метеостанції національних метеорологічних служб. Дані з одночасних або кількох PWS широко використовуються в літературі для широкого кола застосувань і наукових сфер. Це пояснюється різноманітними метеорологічними даними, отриманими PWS, і зростаючою підтримкою з боку наукового співтовариства (Gharesifard et al., 2017). Наприклад, дані PWS використовувалися для дослідження гідрометеорологічних явищ, таких як повінь (Bedrina et al., 2012) або моніторинг опадів (De Vos et al., 2017). Дані PWS були використані як вхідні дані для досліджень енергетичного моделювання (Touzani et al., 2018, Pham et al., 2019) та для досліджень енергоефективності (Kazmi et al., 2018). У сфері відновлюваної енергії та енергетичних систем Desta et al. (2017) моделювали виробництво вітру з використанням даних PWS. Боттаччолі та ін. (2019) досліджували вихід сонячної фотоелектричної енергії за допомогою даних PWS в контексті управління розумними мережами, тоді як Іпанлуганжі та ін. (2018) реалізували прогнозування сонячної генерації з використанням цього джерела даних. У аналізі сонячних ресурсів Лопес Лоренте та ін. (2019) показали, як середньогодинна глобальна горизонтальна освітленість (GHI), отримана від PWS, мала вищу кореляцію з виробництвом фотоелектричних установок, ніж від офіційних метеостанцій [9].

Багато сучасних проектів моніторингу та контролю, таких як системи на фабриках, вдома та інші, використовували Інтернет речей (IoT). Ці пристрої виконують самостійні функції, не вимагаючи ручного втручання, щоб підвищити зручність і безпеку. Електричні мережі є однією з найважливіших областей, у якій системи IoT можуть контролювати, відстежувати, виявляти та попереджати про несправності, оскільки виявляють несправності, відстежують мережеві дані та знаходять найкращі рішення за менший час для підвищення ефективності та надійності електричних мереж.

В [10] представлена модель на основі бездротової сенсорної мережі (WSN). Ця система відстежує та контролює різноманітні електричні та екологічні змінні, включаючи споживання електроенергії, температуру погоди, вологість, полум'я, освітлення та виявлення розривів кабелю в електричних стовпах. Кожен датчик є вузлом і підключається до плати мікроконтролера окремо. Дані, зібрані цими датчиками, відображаються та відстежуються на веб-сторінці та зберігаються в базі даних локального сервера.

У [11] пропонується та показується повна система Smart Greenhouse на основі IoT, яка поєднує моніторинг, оповіщення, хмарне зберігання, автоматизацію та прогнозування захворювань в єдиний, легко розгорнутий пакет. Він постійно відстежує змінні навколишнього середовища, такі як температура, вологість і вологість ґрунту, щоб гарантувати кращий урожай і швидко корекцію в разі погіршення обставин. Також включена вбудована автоматизована система управління зрошенням. Нарешті, для виявлення хвороб за допомогою зображень листків він використовує найефективнішу модель глибокого навчання. Крім того, використовуючи хмарне сховище для оптимізації пам'яті та сховища, міський житель може побудувати теплицю і спостерігати за нею зі свого дому, що дозволить йому вжити коригуючих заходів за потреби.

У статті [12] досліджується ідея створення автоматизованої системи моніторингу кухні з використанням Інтернету речей (IoT) і сенсорів на основі мікроконтролерів. Підхід IoT для моніторингу кухні заохочується через те, що кількість користувачів Інтернету в останні роки зросла в геометричній прогресії завдяки технологічним досягненням у сфері телекомунікацій, а також підвищила рівень життя людей. Ця автоматизована система моніторингу кухні має сенсорний блок (для моніторингу параметрів кухні), який підключений до плати мікроконтролера (Arduino UNO), що зображено на Рис.2.1, а передача значень цього параметра через Інтернет здійснюється

за допомогою NodeMCU 1.0. (Рис. 2.2) Система постачається зі зручним мобільним додатком для моніторингу параметрів кухні, таких як температура (у градусах Цельсія), відсоток вологості, потенційний витік газу, випадкові пожежі на кухні та ваги продуктів.



Рисунок 1.1 – Зображення плати Arduino UNO.

Мобільний додаток розроблено за допомогою онлайн-інструменту MIT APP inventor, а базою даних, яка тимчасово зберігає параметри кухні, є база даних Google Firebase Database. Крім того, мобільний додаток також має функцію керування двома електронними приладами (підключеними до двоканального реле) з будь-якої точки світу.

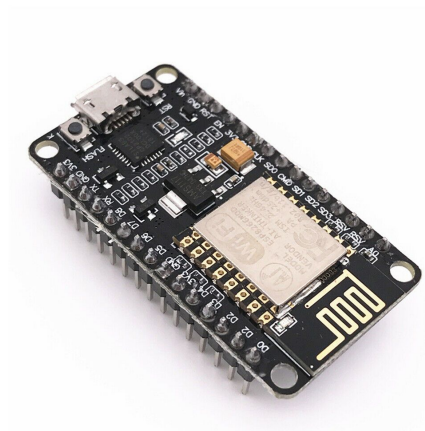


Рисунок 1.2 – Плата NodeMCU ESP8266 v1.0.

В статті [13] описано автоматичну систему моніторингу погоди, яка дозволяє мати динамічні кліматичні дані в реальному часі певної місцевості. Запропонована система базується на технології Інтернету речей та вбудованій системі. Система також включає електронні пристрої, датчики та бездротові технології. Основним завданням цієї системи є визначення параметрів клімату, таких як температура, вологість, наявність деяких газів, на основі датчиків. Отримані значення потім можна надіслати до віддалених програм або баз даних. Після цього збережені дані можна візуалізувати у вигляді графіків та таблиць.

## 2 ОПИС ОБРАНОЇ СИСТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

### 2.1 Постанова задачі

Система повинна являти собою модель пристрою з вимірювання показників навколишнього середовища з такими характеристиками:

- надійність роботи компонентів;
- дешевизна і простота виготовлення;
- точність та актуальність вимірювань;
- можливість зберігання вимірювань для подальшого аналізу статистичних даних;
- можливість сортування та відображення даних у графічному вигляді;
- масштабованість системи;
- можливість удосконалення додаванням датчиків, дисплеїв та програмного забезпечення;

Внаслідок сказаного, проект має складатися з плати з під'єднаними датчиками, яка буде проводити вимірювання показників навколишнього середовища та надсилати їх до бази даних. Після чого має бути можливість відображення даних, для чого буде використовуватися веб-сторінка з таблицями та графіками.

Проект повинен мати можливість тестуватися не лише в дрібному масштабі, але й використовуватися для моніторингу та збору статистичних даних навколишнього середовища для особистих потреб, а також для відповідних закладів та підприємств, яким важливо спостерігати, контролювати, зберігати та аналізувати дані в об'єктах сільського господарства, складських та жилих приміщеннях, закладах з охорони здоров'я, а також на вулиці та природі для зберігання статистики навколишніх показників, які можуть бути використані метеослужбами для

прогнозування зміни погоди та опадів.

Для досягнення поставленої мети мають бути вирішені наступні задачі:

- аналіз систем моніторингу навколишнього середовища;
- вибір апаратної платформи, на якій буде будуватися проект;
- вибір програмного забезпечення для прошивки плати;
- вибір бази даних для зберігання вимірюваних показників;
- вибір та налаштування веб-сервера, на якому буде розташовуватися база даних та фронтенд і бекенд код;
- вибір мов програмування та фреймворків (якщо потрібно) для бекенду та фронтенду;
- вибір хостингу для можливості цілодобового перегляду та використання веб-сторінки у мережі інтернет;
- розробка структурної моделі, апаратної та програмної складової системи;
- розробка веб-додатку для можливості моніторингу та аналізу за допомогою мережі Інтернет;
- завантаження проекту на VPS-хостинг;
- тестування компонентів системи.

Оскільки будь-який проект повинен бути якісно протестованим, то необхідно проконтролювати точність та актуальність вимірювань. Також варто провести тестування різних показників та функціоналу у реальних умовах.

## 2.2 Опис обраної системи

Враховуючи встановлені вимоги до проекту, найкраще підійде система з можливістю бездротової передачі даних. Саме тому було обрано плату NodeMCU на основі мікроконтролера ESP8266, який має можливість підключення до Wi-Fi мережі. (рис. 2.1)

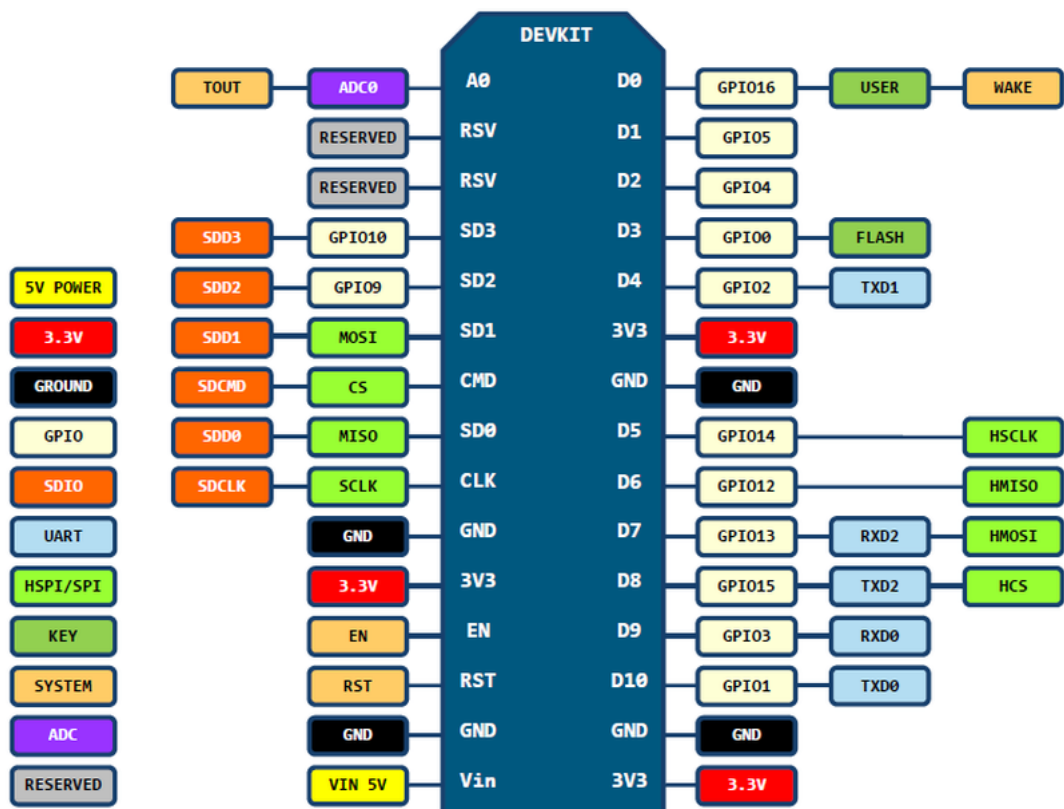


Рисунок 2.1 – Схематичне зображення плати NodeMCU

Характеристики плати NodeMCU V3:

- Wi-Fi 802.11 b/g/n;
- Підтримка STA/AP/STA + AP режимів;
- Вбудований стек протоколів TCP/IP з підтримкою множинних клієнтських підключень (до 5);
- D0 ~ D8, SD1 ~ SD3: можуть бути використані як GPIO, PWM, ІС, тощо;
- Струм на виведення: 15 мА;
- AD0: 1 виведення АЦП;
- Живлення: 4.5-9В (10В максимум), живлення від USB з наданням інтерфейсу відлагодження;

- Споживання: обмін даними:  $\sim 70$  мА (200 мА максимум), очікування:  $< 200$  мкА;
- Швидкість передачі: 110-460800 б/сек;
- Підтримка UART / GPIO інтерфейсів передачі даних;
- Перепрошивка з хмари або через USB;
- Відстань між контактними пінами: 28 мм;
- Діапазон робочих температур:  $-40 \sim +125$  ° C;
- Вага: 18 г. [14]

Модуль BME280 – це черговий датчик тиску фірми Bosch Sensortec для вимірювання атмосферного тиску і температури. У порівнянні з першими датчиками серії (BMP085 і BMP180) він має кращі характеристики та менші розміри – всього 15 мм x 12 мм x 3 мм. (рис. 2.2)

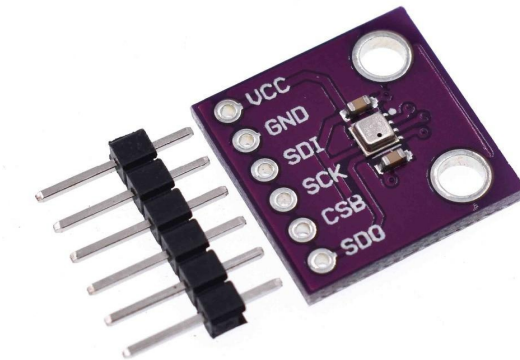


Рисунок 2.2 – Зображення модулю BME280

Характеристики датчика BME280:

- Інтерфейси підключення: SPI, I2C;
- Максимальна швидкодія інтерфейсу: I2C до 3.4МГц;
- Межі вимірювання температури: від -40 до 85 градусів;
- Точність вимірювання температури: від 0.5 до 1 градуса;
- Межі вимірювання вологості: від 0 до 100%;
- Точність вимірювання вологості: 3%;

- Межі вимірювання тиску: від 300 до 1100 гПа;
- Точність вимірювання тиску: 1гПа;
- Напруга живлення: від 1.8 до 5 В;
- Струм в режимі вимірювання тиску: 714 мкА;
- Струм в режимі вимірювання вологості: 340 мкА;
- Споживаний струм в режимі вимірювання температури: 350 мкА;
- Струм в режимі сну: від 0.1 мкА до 0.5 мкА;
- Розміри модулю: 15 x 12 x 3 мм. [15]

Вирішальним аспектом є можливість зберігати зафіксовані параметри.

У даному випадку для цього використовується віддалений веб-сервер. При цьому потрібно не лише зберігати, а і візуалізувати збережені дані. Тому окрім програмування плати та системи передачі параметрів на сервер до бази даних вирішено повноцінний веб-додаток з апаратно-серверною та клієнт-серверною взаємодією. В результаті чого можна буде переглядати на веб-сторінці актуальні та збережені дані у вигляді таблиць та графіків за допомогою мережі інтернет в будь-якій точці світу для зручного моніторингу навколишнього середовища.

## 3 РЕАЛІЗАЦІЯ ПРОЕКТУ

### 3.1 Розробка алгоритмів

Оскільки проект являє собою апаратну платформу з додатковою програмною реалізацією у вигляді веб-додатка, алгоритми розділяються на дві частини: апаратно-серверну та клієнт-серверну.

В апаратно-серверній частині відбувається взаємодія плати з датчиками, підключення до Wi-Fi мережі та відправлення отриманих даних на сервер у вигляді URL-запиту.

Алгоритми клієнт-серверної взаємодії складаються з двох частин: фронтенд (frontend) та бекенд (backend).

Розділення програмних систем на фронтенд і бекенд – одне із стандартних рішень для архітектури програмного забезпечення, пов'язане в програмній інженерії з принципом поділу відповідальності між зовнішнім виглядом та внутрішньою реалізацією. Як правило, бекенд реалізує API, що використовуються фронтендом, і таким чином фронтенд-розробнику не потрібно знати особливості реалізації внутрішньої частини, а бекенд-розробнику – інтерфейсні рішення. Крім того, такий поділ дозволяє використовувати різні інструменти для реалізації внутрішньої та зовнішньої частини системи, ефективніші для відповідних завдань.

#### 3.1.1 Алгоритми апаратно-серверної взаємодії

Алгоритм апаратно-серверної взаємодії включає в себе стадію налаштування та циклічну стадію. В першій стадії відбувається ініціалізація необхідних змінних та підключення до мережі Wi-Fi для подальшої роботи. В другій стадії відбувається зациклення з вказаною затримкою, де виконується

запис даних показників навколишнього середовища з датчика, формування та відправлення запиту на сервер.

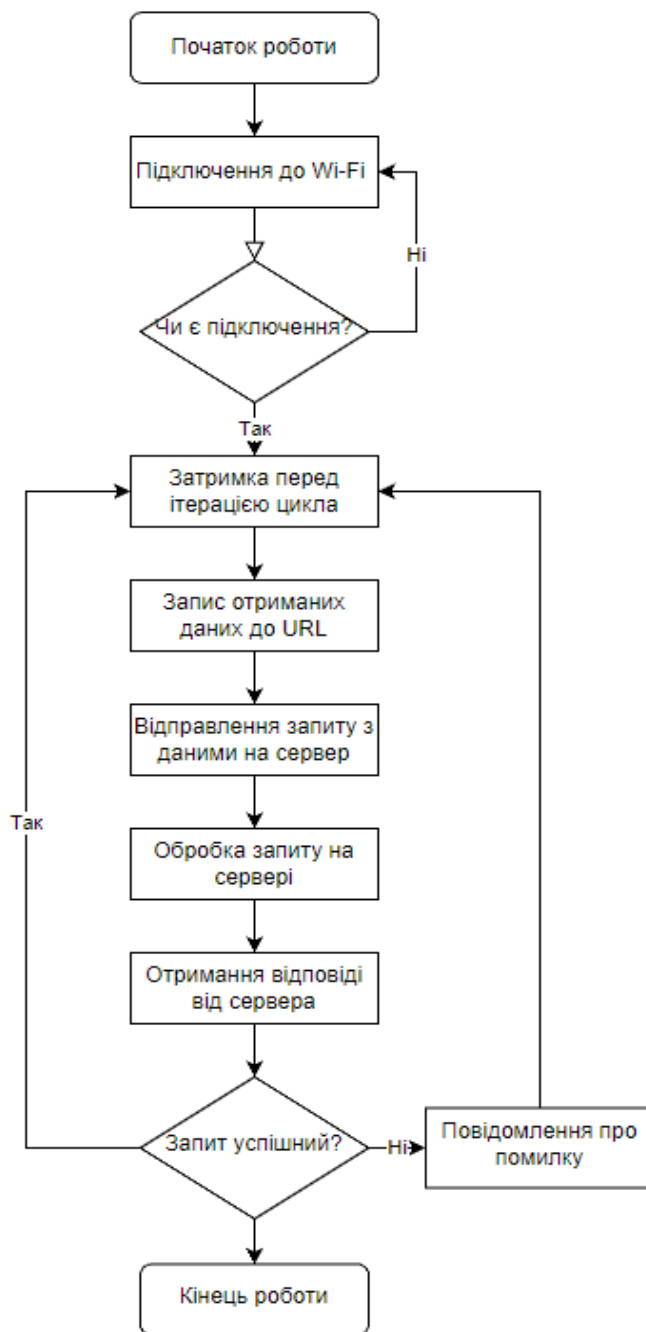


Рисунок 3.1 – Блок-схема алгоритму апаратної частини апаратно-серверної взаємодії

Виконання алгоритму серверної частини апаратно-серверної взаємодії

полягає в отриманні URL-запиту з даними, записі їх до бази даних та відправлення відповіді. Оскільки дані на сервері зберігаються у вигляді таблиць бази даних, то вони будуть виводитись на екран веб-сторінки в двох таблицях: `bme280_current` та `bme280`. Запис до `bme280_current` відбувається при кожному отриманні запиту. Таблиця `bme280_current` містить додаткове поле `max_counter`, яке призначене для реалізації запису даних до таблиці `bme280` після проходження певного інтервалу лічильника. Якщо залишок від ділення `counter` на `max_counter` дорівнює 0, то дані додаються і до таблиці `bme280`. В кінці відправляється відповідь на URL-запит про успішне виконання або помилку (Лістинг 3.4).

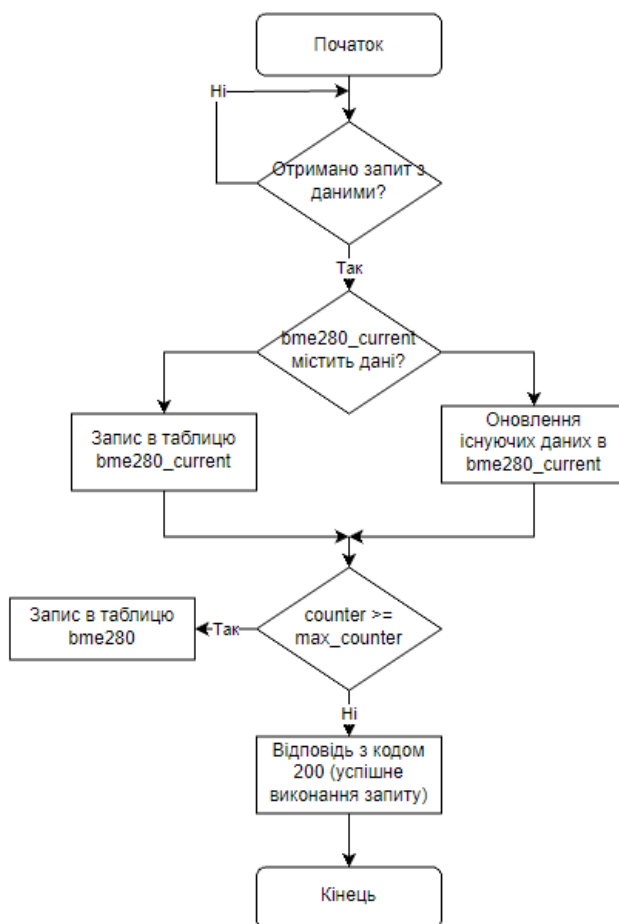


Рисунок 3.2 – Блок-схема алгоритму серверної частини апаратно-серверної взаємодії

### 3.1.2 Алгоритми клієнт-серверної взаємодії

Як вже було згадано, клієнт-серверна взаємодія полягає у тому, що фронтенд формує інтерфейс та функціонал веб-сторінки і за потребою надсилає запити до бекенду, який в свою чергу обробляє отриманий запит (взаємодіє з базою даних, проводить обчислення) та відправляє фронтенду відповідні дані.

Оскільки поставлені цілі потребують лише відображення даних у вигляді таблиць та графіків, то алгоритм їх взаємодії буде полягатися лише у запитах від фронтенду до сервера для отримання даних з двох таблиць бази даних. Для таблиці актуальних даних датчика виконується рекурсійний запит до бекенду, в результаті успішного виконання якого дані, що отримані з таблиці `bme280_current`, відображаються у відповідних полях таблиці веб-сторінки.

Таблиця записів бази даних формується у відповідності до обраних користувачем або вказаним за замовченням значенням сторінки, кількості записів, параметрів сортування, порядку виводу. Запит з перерахованими вище параметрами відправляється до сервера, де виконується вибірка з таблиці `bme280`, та повертаються дані записів таблиці. Ці дані використовуються для заповнення таблиці та графіків, які відображають кожен вимірюваний параметр навколишнього середовища.

### 3.2 Схема пристрою

В роботі використовується I2C шина для підключення датчика BME280 до плати NodeMCU V3 на базі мікроконтролера ESP8266. Для цього було підключено датчик до контактів SDA і SCL плати ESP8266 за допомогою перемичок мама-мама (рис. 2.3):

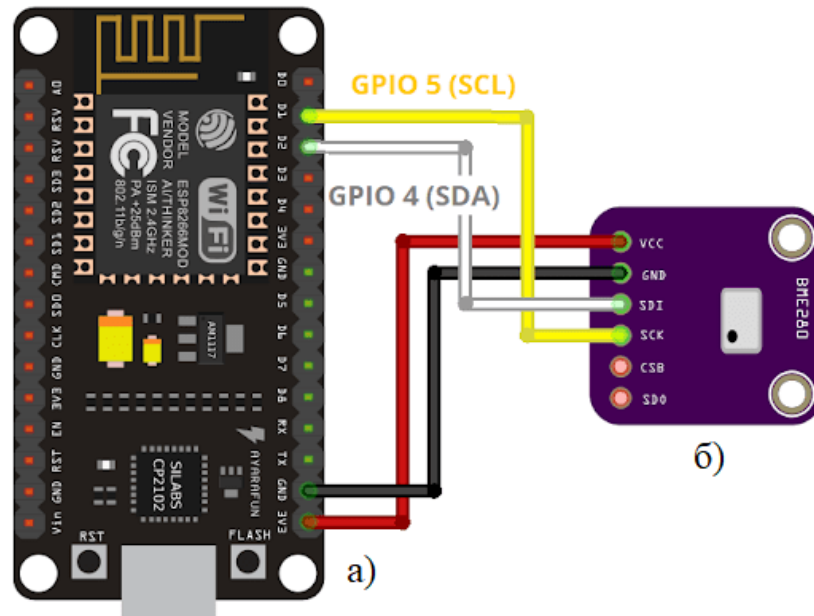


Рисунок 3.3 – Схематичне зображення підключення модулю BME280 до плати NodeMCU за допомогою шини I2C. а) NodeMCU V3; б) BME280

Таблиця 3.1 – Підключення датчика за інтерфейсом I2C

BME280	ESP8266
Vin	3.3V
GND	GND
SCK	GPIO 5
SDI	GPIO 4

Датчик BME280 має три режими роботи:

- Normal mode;
- Forced mode;
- Sleep mode.

Режим Sleep встановлюється за замовчуванням після скидання живлення. У режимі сну вимірювання не проводяться, а споживання енергії мінімальне. Усі реєстри доступні; ідентифікатор мікросхеми та коефіцієнти компенсації можуть бути прочитані. Не існує особливих обмежень щодо таймінгів інтерфейсу.

У режимі Forced – виконується одноразове вимірювання відповідно до

обраних параметрів вимірювання та фільтрації. Після закінчення вимірювання датчик повертається в режим Sleep, і результати вимірювань можна отримати з реєстрів даних. Для наступного вимірювання потрібно знову обрати Forced режим. Використання примусового режиму рекомендується для програм, які вимагають низької частоти дискретизації або синхронізації на основі хоста.

Режим Normal включає автоматичну безперервну циклічність між (активним) періодом вимірювання та (неактивним) періодом очікування [16].

У роботі датчик використовується у режимі Normal, оскільки цього режиму достатньо для поставлених задач.

### 3.3 Реалізація методів

Даний проект було вирішено розробляти з використанням системи контролю версій git та сервісом github для можливості збереження коду на кожному етапі його написання. Весь код проекту можна переглянути в [17], а готову реалізацію веб-додатка на сайті [18].

#### 3.3.1 Скетч Arduino IDE для плати NodeMCU

Використана в апаратній частині платформа NodeMCU V3 зумовлює реалізацію програмної частини у середовищі Arduino IDE із використанням мови програмування, базованої на C++.

Програма повинна мати невелику та просту структуру, для того, щоб швидко оброблятися та вміщуватися на платформі NodeMCU, мати універсальні алгоритми та можливість модифікації для використання з іншими компонентами, а також виконувати поставлені для проекту задачі.

По-перше, у коді проекту підключаються бібліотеки зі спеціальними методами роботи ESP8266 з Wi-Fi, <https> та датчиком BME280 для

повноцінної взаємодії компонентів між собою.

Далі за допомогою директиви `#define` встановлюються константи `LED` (ідентифікатор світлодіоду для індикації відправлення запитів), `DELAY` (затримка перед циклом) та `SEALEVELPRESSURE_HPA` (значення тиску на рівні моря). Такі константи не займають програмну пам'ять, бо компілятор замінює звернення до них на їх значення ще на етапі компіляції. Тому такі директиви використовується для покращення читабельності коду програми. Константа `SEALEVELPRESSURE_HPA` має значення 1013.25 та зберігає значення тиску на рівні моря в гектопаскалях (еквівалентно мілібару). Воно використовується для оцінки висоти для даного тиску шляхом порівняння його з тиском на рівні моря. У цьому прикладі використовується значення за замовчуванням, але для більш точних результатів можна замінити його поточним тиском на рівні моря. Для України використовується Балтійська система висот від нуля кронштадського футштока.

Після цього ініціалізуються змінні назви мережі, паролю, адреси веб-сервера, лічильника, а також температури, тиску, висоти над рівнем моря та вологості.

Кожен скетч для ESP8266 містить в собі дві головні функції: `setup()` та `loop()`. Всередині функції `setup()` виконується ініціалізація послідовного зв'язку та перевірка ініціалізації датчика. Також створюється підключення до Wi-Fi мережі за допомогою функції `WiFi.begin(ssid, password)`, після успішного виконання якої виводиться повідомлення. Після того, як ESP8266 підключається до мережі, скетч виводить IP-адресу, призначену ESP8266, надсилаючи значення результату методу `WiFi.localIP()` в монітор послідовного порту. (Лістинг 3.1)

Лістинг 3.1 – Функція `setup()`

```
void setup() {
  Serial.begin(115200);           // Ініціалізація послідовного зв'язку
```

```

pinMode(LED, OUTPUT);           // Ініціалізація світлодіоду
if (!bme.begin(0x76)) {         // Перевірка ініціалізації датчика
  Serial.println("Could not find a valid BME280 sensor!");
  while (1);
}

Serial.print("Connecting to " );
WiFi.begin(ssid, password);     // Підключення до Wi-Fi мережі
while (WiFi.status() != WL_CONNECTED)
{
  delay(500);
  Serial.print(".");
}
Serial.println(" connected");
Serial.println("IP address: "); // Вивід локальної IP-адреси
Serial.println(WiFi.localIP()); // до Serial порту
}

```

У функції `loop()` (Лістинг 3.2) відбувається зчитування з датчика та запис у відповідні змінні даних температури, тиску, вологості повітря і висоти над рівнем моря, а також їх вивід до Serial-порту. Далі у випадку наявності з'єднання з Wi-Fi мережею викликається функція `sendData()` (Лістинг 3.3), в яку передаються вищеперечислені змінні для відправлення https запиту на веб-сервер.

### Лістинг 3.2 – Функція `loop()`

```

void loop() {
  delay(DELAY);                // Затримка перед ітерацією циклу
  digitalWrite(LED, LOW);      // Увімкнення світлодіоду
  temp = bme.readTemperature(); // Зчитування параметрів з датчика
  pressure = bme.readPressure()/100.0F; // приведення до гПа,
  alt = bme.readAltitude(SEALEVELPRESSURE_HPA);
  hum = bme.readHumidity();
  Serial.println("\nTemperature: "+(String)temp); // Вивід даних до Serial-порту
  Serial.println("Pressure: "+(String)pressure);
  Serial.println("Altitude: "+(String)alt);
  Serial.println("Humidity: "+(String)hum);
  Serial.println("Counter: "+(String)counter);
  if (WiFi.status() == WL_CONNECTED) // Перевірка підключення до Wi-Fi
    // Виклик функції відправки запиту з даними на веб-сервер
    sendData(temp, pressure, alt, hum, counter);
  digitalWrite(LED, HIGH);     // Вимкнення світлодіоду
  counter >= 100000 ? counter=1 : counter++; // Збільшення або скидання
  лічильника за допомогою тернарного оператора
}

```

### Лістинг 3.3 – Функція sendData()

```

void sendData(float temp, float pressure, float alt, float hum, int counter) {
    WiFiClientSecure client;
    client.setInsecure(); // Використання безпечного протоколу (https)
    HTTPClient https;
    String data = "/src/insert.php?temp="+String(temp)+ // Заповнення строки даних
"&press="+String(pressure)+
"&alt="+String(alt)+
"&hum="+String(hum)+
"&counter="+String(counter);
    String fullUrl = host + data; // Конкатенація строк адреси хоста та даних
    Serial.println("Requesting " + fullUrl); // Вивід до Serial порта URL запиту
    if (https.begin(client, fullUrl)) {
        int httpCode = https.GET(); // Відправлення URL запиту
        Serial.println("Response code: " + String(httpCode)); // Вивід коду відповіді
        if (httpCode>0) // Вивід відповіді на запит
            Serial.println("Response: " + https.getString());
        https.end(); // Завершення роботи протоколу
    } else { // Вивід повідомлення при помилці
        Serial.printf("[HTTPS] Unable to connect\n");
    }
}
}

```

#### 3.3.2 Файлова структура проекту

Для того, щоб структурувати та розділяти функціонал фронтенду та бекенду, мати можливість використовувати конфіденціальну інформацію (логіни, паролі бази даних і т.д.) було вирішено створити типову файлову структуру веб-додатка з використанням мови PHP (Рис. 3.4). Вона поділяється на такі папки та файли:

– папка `config/` використовується для бекенду та містить конфігураційний файл `config.js`, який зберігає в собі константи, використані в кодї (заголовки таблиць, значення параметрів для сортування, список міст для відображення погоди і т.д.), а також файл `.env.php`, в якому прописані параметри бази даних та токен для відправлення запиту погоди, який не індексується `git`'ом для запобігання взлому бази даних сервера або витоку

персональних даних;

– папка `public/` містить 3 підпапки (`css`, `img`, `js`) для фронтенду, в яких зберігається код та картинки, необхідні для відображення та функціонування веб-сторінки.

– папка `src/` зберігає бекенд `.php` файли, які використовуються для отримання запитів, їх обробку, взаємодію з базою даних за допомогою MySQL запитів тощо.

В корені проекту зберігається `index.php`, який перенаправляє в `index.html` файл, де відбувається побудова компонентів веб-сторінки та виклик їх основних функцій. Також в корені є допоміжні файли, такі як `.gitignore` (визначення файлів, які `git` буде ігнорувати), `.htaccess` (конфігураційний файл сервера), `*.sql` (експортована база даних), `readme.md` (файл, в якому описано проект для `github`)

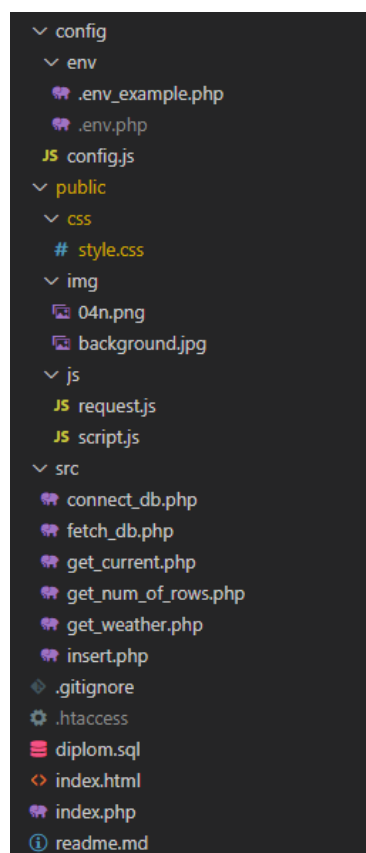


Рисунок 3.4 – Файлова структура проекту

### 3.3.3 Реалізація бекенд коду веб-додатку

Як вже згадувалось раніше, функціонування веб-додатку виконується за допомогою мови PHP для роботи на сервері (бекенду) та HTML, CSS і JavaScript для роботи зі сторони клієнта (фронтенд).

Бекенд містить .php файли, які використовуються в запитах від плати (insert.php) та фронтенду для отримання змінних та їх використання у роботі з базою даних. Так, наприклад, файл insert.php (Лістинг 3.4), до якого звертається плата за допомогою GET-запиту, записує отримані дані в таблицях bme280\_current та bme280 (якщо значення лічильника, що передане з плати, кратне значенню поля max\_count з таблиці), та відправляє отриманий результат у вигляді значення змінної max\_count.

#### Лістинг 3.4 – Файл insert.php

```
<?php
include "connect_db.php"; // З'єднання з файлом connect_db.php
if (isset($_GET['temp'])) // Ініціалізація змінних з URL-запиту
    $temp = $_GET['temp']; // $temp для температури
if (isset($_GET['press']))
    $press = $_GET['press']; // $press для тиску
if (isset($_GET['alt']))
    $alt = $_GET['alt']; // $alt для висоти
if (isset($_GET['hum']))
    $hum = $_GET['hum']; // $hum для вологості
if (isset($_GET['counter']))
    (int)$counter = $_GET['counter']; // $counter для лічильника
// Виконання запиту до БД для перевірки наявності даних в таблиці bme280_current
$query = $conn->query("SELECT * FROM bme280_current");
if(mysqli_num_rows($query) != 0) { // Якщо в bme280_current є дані
    $sql = "UPDATE bme280_current SET temp_bme280 = $temp, press_bme280 =
$press, alt_bme280 = $alt, hum_bme280 = $hum";
    mysqli_query($conn, $sql); // Оновлення даних в bme280_current
} else {
    $sql = "INSERT INTO bme280_current (temp_bme280, press_bme280, alt_bme280,
hum_bme280, max_count) VALUES ($temp, $press, $alt, $hum, 10)";
    mysqli_query($conn, $sql); // Внесення даних до таблиці
}
// Виконання запиту до БД для зчитування значення max_count з таблиці bme280_current
$result = $conn->query("SELECT max_count FROM bme280_current LIMIT 1");
$row = $result->fetch_array(MYSQLI_ASSOC);
(int)$maxcount = $row["max_count"];
```

```
// Внесення даних до таблиці bme280 якщо $counter кратний $maxcount
if ($counter % $maxcount == 0) {
    $sql = "INSERT INTO bme280 (temp_bme280, press_bme280, alt_bme280, hum_bme280)
VALUES ($temp, $press, $alt, $hum)";
    mysqli_query($conn, $sql);
}
echo $maxcount; // Повернення значення $maxcount
?>
```

Файл `connect_db.php` викликається в кожному `.php` файлі, де потребується взаємодія з базою даних. Він виконує підключення до БД за допомогою функції `mysqli()`, до якої передаються параметри бази даних, які отримані з конфігураційного файлу `.env.php`.

### Лістинг 3.5 – Файл `connect_db.php`

```
<?php
include "../config/env/.env.php"; // Підключення файлу .env.php
$conn = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME); // Підключення до БД
if (!$conn) // Повідомлення при неможливості підключення до БД
    die("Помилка: не вдається підключитися: " . $conn->connect_error);
?>
```

Принцип реалізації в інших файлах бекенду схожий на `insert.php`: отримується запит з параметрами, проводиться вибірка з бази даних, повертається результат (Додаток В).

Різниця є лише в файлі `get_weather.php` (Лістинг 3.6): там замість звернення до бд виконується запит до API-сервісу `api.openweathermap.org`, який дозволяє отримати актуальну інформацію про погоду для будь-якого міста, в даному випадку це один з обраних обласних центрів України. Для виконання цього підключається вже згаданий файл `.env.php`, в якому у змінній `$APPID` зберігається значення токена ідентифікатора користувача. В параметрах запиту вказані: місто, токен користувача, система виміру – метрична, мова результату – українська. Дані повертаються у форматі `json`.

### Лістинг 3.6 – Файл get\_weather.php

```
<?php
include "../config/env/.env.php"; // Підключення файлу зі змінною $APPID
if (isset($_GET['city'])) // Зчитування змінної city з URL
    $city = $_GET['city'];
// Запит до API для отримання даних погоди обраного міста у форматі json
$result=file_get_contents("https://api.openweathermap.org/data/2.5/weather?q=$city&appid=$APPID&units=metric&lang=ua");
echo $result; // Повернення отриманих даних погоди
?>
```

#### 3.3.4 Реалізація фронтенд коду веб-додатку

Фронтенд код веб-додатку складається з HTML, CSS та JavaScript коду [17]. index.html є головним файлом, в заголовку якого виконується підключення інших файлів, бібліотек тощо. В ньому у прописана структура елементів, відображених на веб-сторінці, а також поле скрипт, в якому викликаються функції з файлів script.js та request.js (Додаток Б), які створюють та заповнюють поля, таблиці, графіки при відкритті сторінки.

style.css є файлом формату .css, який розшифровується як «Cascading Style Sheets», тобто каскадні таблиці стилей. В цьому файлі визначається стилізація вказаних в .html файлі елементів (колір, розмір, фон, поведінка тощо).

В файлах script.js та request.js (Додаток В) виконується функції, які написані мовою JavaScript. В request.js визначені функції, які за допомогою асинхронних ажах-запитів відправляють запити до сервера для динамічного отримання та заповнення елементів сторінки без потреби її перезавантаження.

В файлі script.js визначаються функції, які викликаються для зміни стану елементів сторінки: при першому запуску виконується зчитування з локального сховища змінних параметрів, які були обрані та збережені під час минулого користування (місто, параметри сортування, відкрита сторінка

таблиці), та заповнення таблиць та графіків, а також функції, які викликаються у відповідь на дії користувача. Так, наприклад, після вибору користувачем іншого значення параметра сортування відбувається виклик функції `changeValue(name, value)`, яка записує обране значення в локальне сховище браузера та викликає функцію `fetchDB()` з файла `request.js`, яка зчитує усі обрані параметри та надсилає аґах-запит до файла `fetch_db.php` на сервері, після отримання результату від якого за допомогою бібліотеки `jQuery` заповнює таблицю та графіки отриманими значеннями.

Лістинг 3.7 – Функції файлів `script.js` та `request.js` для оновлення значень таблиці «База даних»

```
// Функція обробки натискання на елемент сортування та оновлення таблиці "База даних"
function changeValue(itemName, value) {
    localStorage.setItem(itemName, value);
    fetchDB();
}
function fetchDB() { // Функція, яка за допомогою аґах-запиту до файла
    var items = getItems(); // "src/fetch_db.php" отримує дані таблиці
    $.ajax({ // в обраних межах та за відповідними умовами
        type: "GET", // після чого викликає функції створення таблиці бази даних та графіків
        url: "/src/fetch_db.php?" + "page=" + items.page + "&count=" + items.count + "&param=" +
        items.param + "&order=" + items.order,
        dataType: "json",
        success: function (result) {
            for (let index = 0; index < result.length; index++) {
                var object = result[index];
                for (const key in object) {
                    if (Object.hasOwnProperty.call(object, key)) {
                        if (key == "date_bme280") // Розділення строки на дату та час
                            object[key] = object[key].split(" ");
                    }
                }
                result[index] = object;
            }
        } // Функція для створення таблиці "База даних" з оновленими даними
        var $table = createTable(result, DB_HEADER);
        $("#dbTable").empty();
        $table.appendTo($("#dbTable")); // Заміна попередньої таблиці на оновлену
        var res = fetchResult(result); // Вибірка з результатів даних для кожного графіка
        drawCharts(res, res.date); // Функція заповнення графіків
    },
    error: function (jqXHR, exception) { // Повідомлення у випадку помилки
        printError(jqXHR, exception, '#post');
    },
});};
```

## 4 ТЕСТУВАННЯ

### 4.1 Постанова задач на тестування

Розроблена система має бути якісно протестована у різних умовах та ситуаціях як на апаратному рівні, так і на програмному.

В ході тестування повинні змінюватись та аналізуватись усі параметри навколишнього середовища, які вимірюються датчиками.

Не менш важливою частиною є тестування функціональності веб-сторінки. Необхідно, щоб усі елементи завантажувалися та відображалися коректно, або у випадку несправності однієї з систем виводилося повідомлення про помилку.

У ході тесту пристрій поміщується в стартові умови, після чого ці умови будуть змінюватися. Збережені у базі даних показники можна буде переглянути у відповідній таблиці та графіках на веб-сторінці, обравши необхідну кількість виводимих значень та перейшовши на потрібну сторінку, коли ці значення були зафіксовані.

Для проведення тестування був визначений набір типових ситуацій, що необхідно протестувати:

- Вимірювання у незмінюваних кімнатних умовах впродовж довгого часу;
- Вимірювання в холодних умовах (зимою є можливість тестування на балконі);
- Вимірювання в теплих умовах;
- Тестування з різним значенням інтервалу збереження даних;
- Вибір різних міст для відображення погоди;
- Тестування меню кількості виводимих рядків;
- Вибір різних значень навігатора сторінок;

- Вибір різних параметрів сортування;
- Вибір різних значень порядку сортування;
- Взаємодія з графіками;

## 4.2 Тестування процесу вимірювання показань в різних умовах

### 4.2.1 Тестування в статичних кімнатних умовах

Обравши вивід 1000 значень та 7 сторінку при сортуванні по ід по порядку, можна побачити результати вимірювань впродовж ночі з 20:42 10.12.2022 по 6:40 11.12.2022. По графіку температури можна помітити, що температура з 20,7 °С в 20:42 піднялася та трималася на рівні 21,6 °С до 23:30, після чого опустилася в середньому до 19,9 °С та трималася так до 6:40 ранку. За цим можна зробити висновок, що вимірювання температури в статичних умовах відбувається з погрішністю в 0,15-0,20 °С, хоча в документації датчика повідомлюється про погрішність вимірювань 0,5-1,0 °С через нагрівання самого датчика та його компонентів.

Зменшення показників атмосферного тиску протягом вимірюваного часу з 965 гПа до 957 гПа зумовлено спочатку нагрітим повітрям, а після півночі інформацією, що протягом доби він двічі підвищується (вранці та ввечері) і двічі знижується (після полудня та після півночі).

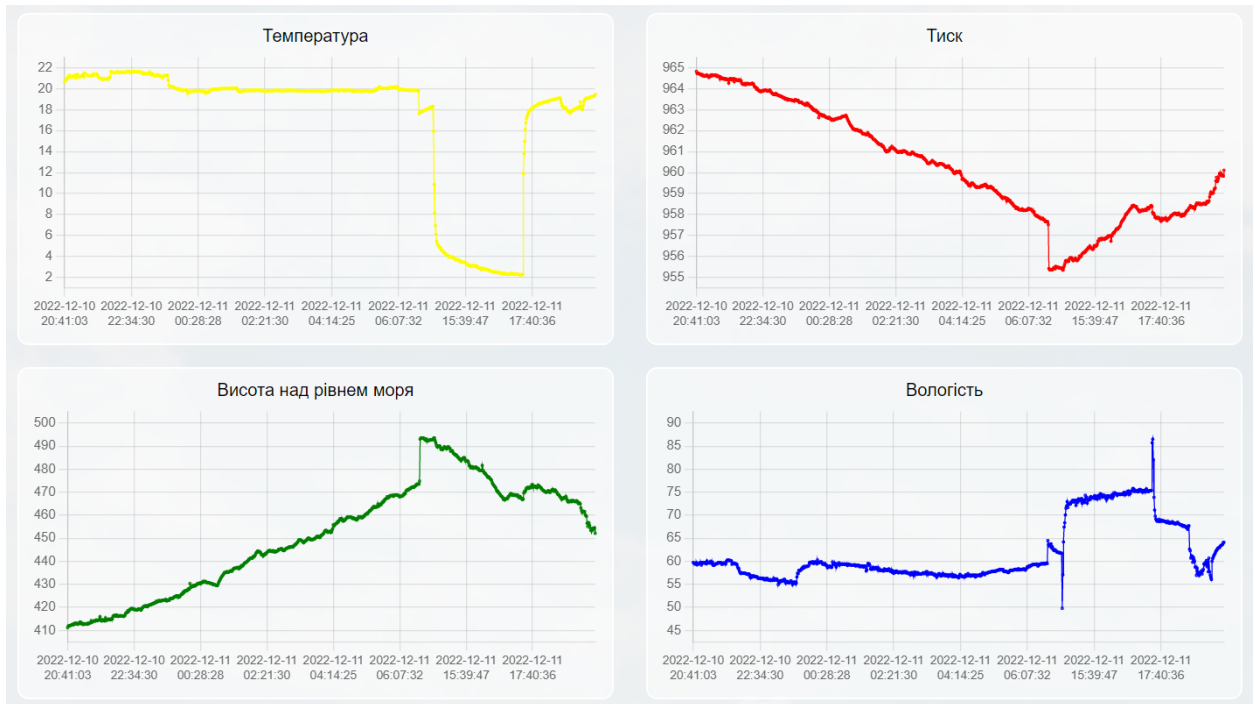


Рисунок 4.1 – Графіки вимірювань в статичних та динамічних умовах

#### 4.2.2 Тестування в умовах низької температури

Дані вимірювань в умовах, близьких до вуличних, можна побачити на попередніх графіках, або обравши вивід 500 значень та сторінку 14 при сортуванні по id по порядку. Тут одразу можна помітити, що з 14:40 по 17:24 температура повітря знизилася спочатку до 5 °С, після чого продовжила падати до 2,25 °С (17:20). Після зміни умов навколишнього середовища точність вимірювання не можна гарантувати через фактор нагрівання або охолодження поверхні датчика. Тиск відповідно в цей проміжок часу збільшувався, оскільки чим нижче температура, тим більше буде тиск. Вологість одразу після зміни умов по зросла та перебувала в діапазоні від 72% до 75%, що відповідає вказаній в характеристиках датчика погрішності вимірювання вологості повітря у 3%.

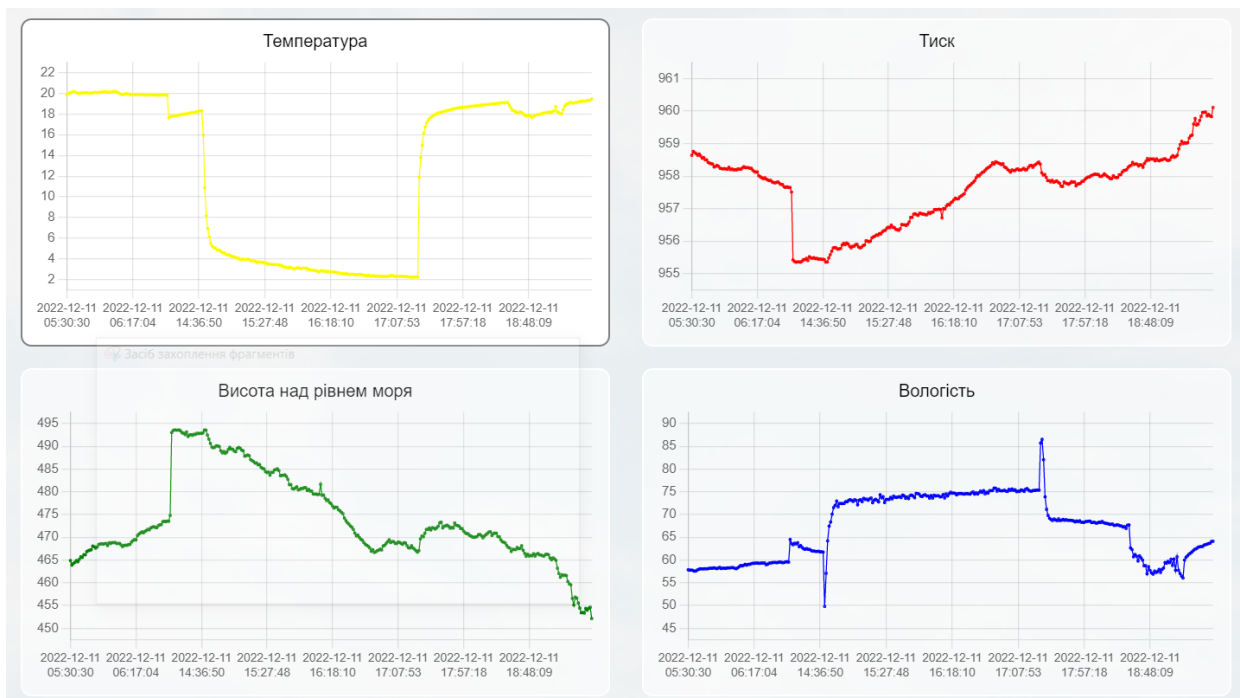


Рисунок 4.2 – Графіки вимірювань в умовах низької температури

#### 4.2.3 Тестування в умовах високої температури

Після поміщення датчика в теплі умови навколишнього середовища, а саме під включений обігрівач, на 138 сторінці для 50 записів по зростанню на графіках відображено поступове збільшення температури з 18 °С до 35,2 °С (Рис. 4.3). Таке збільшення температури зумовлено поступовим нагріванням повітря навколо самого обігрівача в процесі його роботи. На показники тиску це не вплинуло, а ось графік вологості зі значень близько 68% опустився до 35-36 °С. Також по цих графіках можна побачити залежність вологості від температури в даних умовах.



Рисунок 4.3 – Графіки вимірювань в умовах високої температури

#### 4.2.4 Тестування з різним значенням інтервалу збереження даних

Керування частотою запису отриманих значень в таблицю `bme280` здійснюється за допомогою зміни значення поля `max_count` в таблиці `bme280_current` (Рис. 4.1). Так, наприклад, стандартне значення 10 зумовлює запис до таблиці `bme280` кожне 10 вимірювання, що дорівнює приблизно одній хвилині.

time	max_count	temp_bme280	press_bme280	alt_bme280	hum_bme280
2022-12-11 19:12:00	10	18.74	958.72	464.21	57.39

Рисунок 4.4 – Структура таблиці `bme280_current`

Вручну змінивши значення `max_count` на 50 в 19:08, інтервал між записами в таблиці збільшився відповідно від 65-75 секунд до 340-400

секунд. Інтервал після зміни виділено жовтим.

ID	Дата	Час	Температура	Тиск	Висота	Вологість
6835	2022-12-11	19:42:26	18.42	959.03	461.56	59.99
6834	2022-12-11	19:36:21	18.03	959	461.76	56.05
6833	2022-12-11	19:30:02	18.01	959.07	461.21	56.46
6832	2022-12-11	19:23:43	18.14	958.97	462.03	57.03
6831	2022-12-11	19:17:29	18.25	958.84	463.19	57.74
6830	2022-12-11	19:11:40	18.75	958.64	464.92	60.69
6829	2022-12-11	19:07:56	18.32	958.6	465.26	57.74
6828	2022-12-11	19:06:46	18.25	958.58	465.48	60.18
6827	2022-12-11	19:05:37	18.13	958.63	465.05	58.82
6826	2022-12-11	19:04:31	18.21	958.58	465.44	60.17

Рисунок 4.5 – Дані таблиці bme280 після зміни значення max\_count в таблиці bme280\_current

### 4.3 Тестування функціоналу веб-сторінки

Веб-сторінка має чотири блоки:

- блок прогнозу погоди для обраного міста в заголовку (Додаток Б)
- таблиця «Дані датчика BME280» для відображення актуальної інформації з сервера
- таблиця «База даних» та елементи сортування
- графіки

Користувач може взаємодіяти з усіма блоками, окрім таблиці «Дані датчика BME280». Загальний вигляд інтерфейсу сторінки в різних умовах є в Додатку Б.

### 4.3.1 Тестування відображення погоди різних міст

Для тестування точності відображення актуальної погоди в містах було обрано місто Львів та Кропивницький. Дані порівнювалися з інформацією на сайті українського гідрометеорологічного центру [19].

Для обох міст дані з API та гідрометеорологічного центру співпадають у межах погрешностей (Рис. 4.6). Різниця лише в відображенні стану погоди для Кропивницького: в API «хмарно», а на сайті гідрометеорологічного центру «дощ».

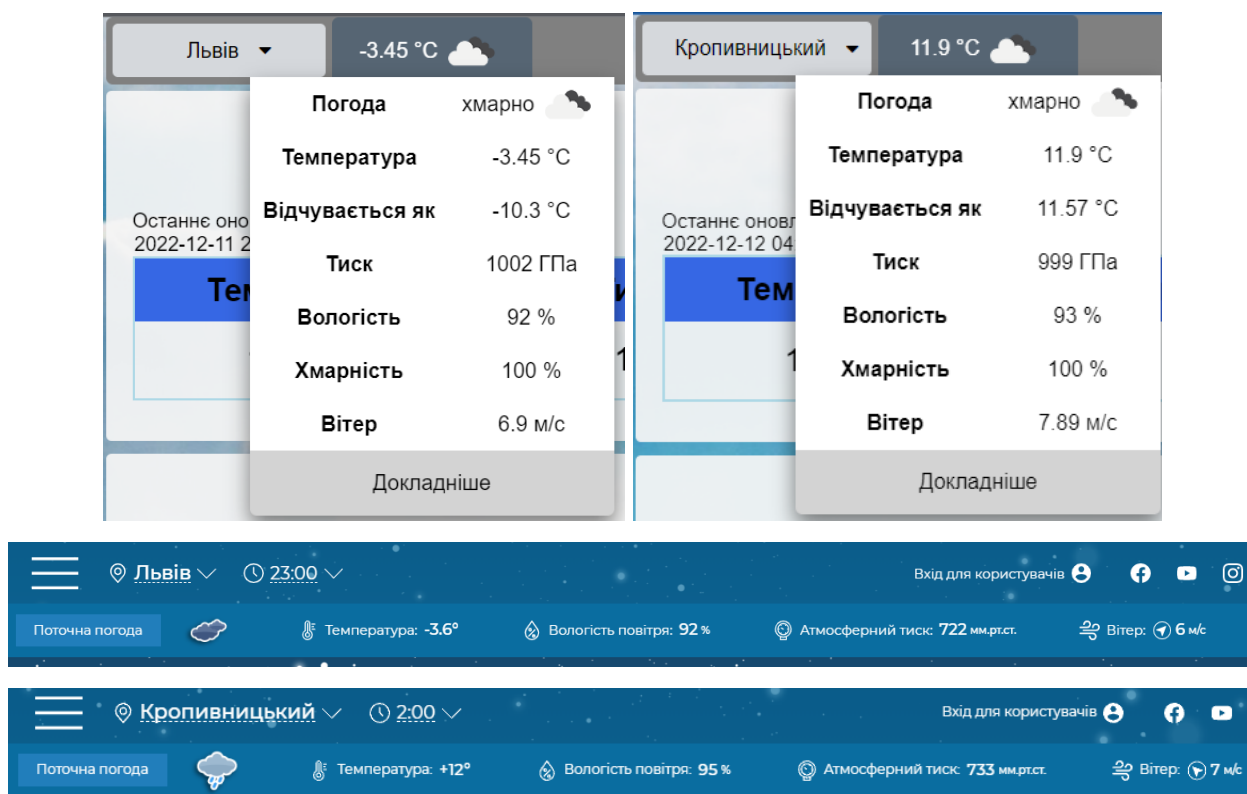


Рисунок 4.6 – Порівняння інформації про погоду для Львова та Кропивницького

### 4.3.2 Тестування параметрів сортування

Для тестування параметрів сортування було обрано такі показники: кількість

рядків: 500, сортування: по температурі, порядок: по зростанню, сторінка: 2.  
В результаті на рис. 4.7 можна побачити, що відображені дані відповідають заявленим потребам.

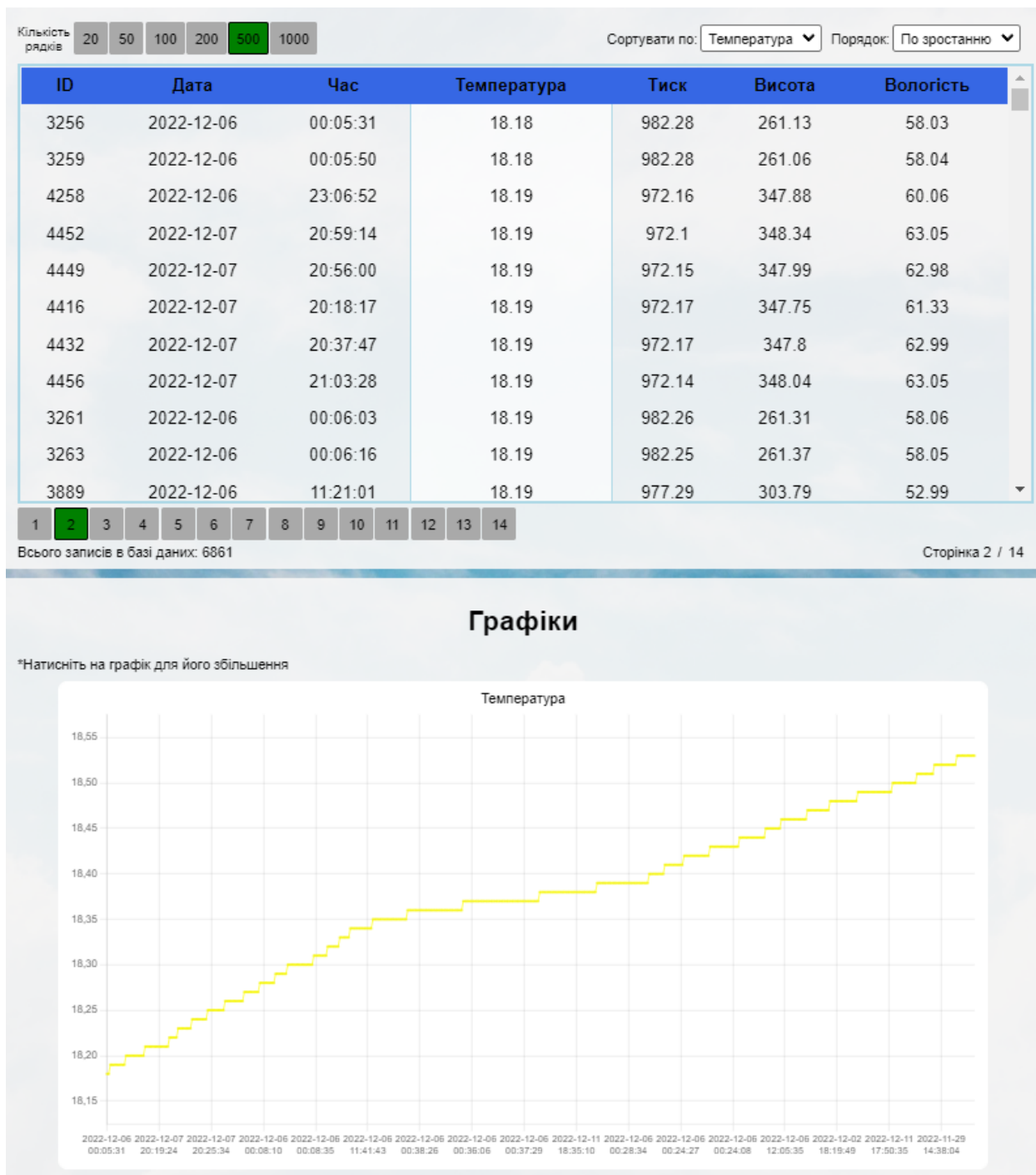


Рисунок 4.7 – Тестування параметрів сортування

### 4.3.3 Тестування графіків

На рис. 4.8 показаний приклад того, як можна взаємодіяти з графіками. Було натиснуто на графік температури та вологості, а також наведено мишкою на одне зі значень графіка температури.



Рисунок 4.8 – Тестування графіків

#### 4.4 Результати тестування

Після проведення усіх тестів, можна зробити такі висновки:

– датчик справно вимірює показники навколишнього середовища в статичних умовах відповідно до зазначених в технічних характеристиках погрешностей.

– датчик справно реагує на зміни умов навколишнього середовища: температуру, тиск, вологість;

– веб-сервер швидко реагує та оновлює інформацію відповідно до обраних параметрів;

– API сервіс для відображення погоди в зазначеному місті надсилає дані, відповідні до даних на офіційному українському гідрометеорологічному центрі.

## ВИСНОВКИ

В результаті виконання роботи розглянуто сутність та проблеми моніторингу навколишнього середовища, порівняно декілька різноманітних систем зі збору та аналізу даних на основі моніторингу кліматичних показників навколишнього середовища та розроблено власну систему.

Під час виконання роботи були поставлені та виконані такі завдання:

- виконано аналіз систем моніторингу навколишнього середовища;
- виконано аналіз IoT систем моніторингу навколишнього середовища;
- розроблено систему моніторингу навколишнього середовища, веб-додаток до неї на основі мов php, html, css, javascript;

В кінцевому результаті можна зробити висновок, що системи моніторингу на основі мікроконтролера дозволять мінімізувати часові та матеріальні витрати для організації виконуваних задач, а також створювати інноваційні науково-технічні рішення соціальних та екологічних проблем світової спільноти.

Дані таких систем вже використовувалися для досліджень, моделювання, аналізу енергетичних систем та установок, тому їх практична значимість буде тільки збільшуватись.

Технологія Інтернету речей стрімко розвивається і вже обійшла звичайні системи за характеристиками та функціями. Її застосування у сфері віддаленого моніторингу та розширеної аналітики мають значні переваги для різного роду бізнесу.

Система IoT відсилає миттєві сповіщення в режимі реального часу. Таким чином, система моніторингу дозволяє відстежувати параметри навколишнього середовища на веб-/мобільній платформі. Це дозволяє уникнути такої роботи, як зняття показань вручну, заощаджуючи час і прискорюючи прийняття рішень.

Дані, зібрані з датчиків температури, можна використовувати для створення статистичних даних. Ця інформація включатиме тривалість часу і деталі, пов'язані з показниками температури. Це допоможе компаніям підвищити надійність своїх складів і холодильних камер.

Оскільки IoT включає розширені можливості телеметрії, дистанційний моніторинг температури також є доступним. Доступ до даних, зібраних за допомогою датчиків моніторингу температури, можна отримати з віддалених місць у веб-додатку або мобільному додатку.

Основними перевагами систем моніторингу на основі Інтернету речей є:

- розумна система моніторингу погоди досить мала та може бути легко встановлена порівняно зі звичайною системою моніторингу погоди. Вимоги до живлення для розумних систем моніторингу погоди досить низькі;

- датчики, які використовуються в розумних системах моніторингу погоди, є значно дешевшими, що робить ці проекти досить економічно ефективними;

- дані з датчиків також можна надсилати на веб-сторінку, до якої можна отримати доступ з будь-якої точки світу.

Головним недоліком таких систем є те, що вони значною мірою покладаються на Інтернет і не можуть ефективно функціонувати без нього.

Використовуючи технологію IoT, інші системи, такі як бездротові сенсорні мережі та системи керування, можуть працювати розумніше та ефективніше.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. "Створення ефективної системи моніторингу довкілля в Україні: проблеми і шляхи їх вирішення". Аналітична записка [Електронний ресурс]. – Режим доступу: [www / URL: http://old2.niss.gov.ua/articles/2054/](http://old2.niss.gov.ua/articles/2054/)
2. Кліматичний моніторинг і його завдання [Електронний ресурс]. – Режим доступу: [www / URL: http://www.novaecologia.org/voecos-1053-1.html](http://www.novaecologia.org/voecos-1053-1.html)
3. Білецька Г.А. - Моніторинг довкілля [Електронний ресурс]. – Режим доступу: [www / URL: https://lubbook.org/book\\_571\\_glava\\_1\\_Tema\\_1.\\_Priroda\\_b%D1%96znesu\\_%D1%96\\_jj.html](https://lubbook.org/book_571_glava_1_Tema_1._Priroda_b%D1%96znesu_%D1%96_jj.html)
4. Білецька Г.А. - Моніторинг довкілля [Електронний ресурс]. – Режим доступу: [www / URL: https://lubbook.org/book\\_571\\_glava\\_8\\_Tema\\_8.\\_Derzhavna\\_re%D1%94stra%D1%81%D1%96.html](https://lubbook.org/book_571_glava_8_Tema_8._Derzhavna_re%D1%94stra%D1%81%D1%96.html)
5. Метеостанція [Електронний ресурс]. – Режим доступу: [www / URL: https://ru.wikipedia.org/wiki/Метеостанция](https://ru.wikipedia.org/wiki/Метеостанция)
6. Какой должна быть лучшая метеостанция для дома [Електронний ресурс]. – Режим доступу: [www / URL: http://ventilationpro.ru/estestvennaya-ventilyatsiya/kakojj-dolzhna-byt-luchshaya-meteostanciya-dlya-doma.html](http://ventilationpro.ru/estestvennaya-ventilyatsiya/kakojj-dolzhna-byt-luchshaya-meteostanciya-dlya-doma.html)
7. Temperature & Environment Sensors [Електронний ресурс]. – Режим доступу: [www / URL: https://avtech.com/Products/Sensors/](https://avtech.com/Products/Sensors/) – Загол. з екрану.
8. Wireless Portable Microcontroller based Weather Monitoring Station. – 18 December 2014 – <https://doi.org/10.1016/j.measurement.2015.08.027>

9. Worldwide evaluation and correction of irradiance measurements from personal weather stations under all-sky conditions. – 28 January 2020. – <https://doi.org/10.1016/j.solener.2020.06.073>
10. Hassan, J.A., Jasim, B.H. Design and implementation of internet of things-based electrical monitoring system (2021) Bulletin of Electrical Engineering and Informatics, 10 (6), pp. 3052-3063. – <https://beei.org/index.php/EEI/article/view/3155>
11. Pothuganti, K., Sridevi, B., Seshabattar, P. IoT and Deep Learning based Smart Greenhouse Disease Prediction (2021) 2021 6th International Conference on Recent Trends on Electronics, Information, Communication and Technology, RTEICT 2021, pp. 793-799. – <https://ieeexplore.ieee.org/document/9573794>
12. Umapathi, N., Sabbani, S. An Internet of Things (IoT)-based Approach for Real-Time Kitchen Monitoring Using NodeMCU 1.0 (2022) Lecture Notes in Electrical Engineering, 792, pp. 35-43 – [https://link.springer.com/chapter/10.1007%2F978-981-16-4625-6\\_4](https://link.springer.com/chapter/10.1007%2F978-981-16-4625-6_4)
13. Mabrouki, J., Azrou, M., Dhiba, D., Farhaoui, Y., Hajjaji, S.E. IoT-based data logger for weather monitoring using arduino-based wireless sensor networks with remote graphical application and alerts (2021) Big Data Mining and Analytics, 4 (1), art. no. 9321200, pp. 25-32. – <https://ieeexplore.ieee.org/document/9321200>
14. Wi-Fi модуль NodeMCU V3 ESP8266 (CH340) [Электронный ресурс]. – Режим доступа: [www / URL: https://arduino.ua/prod1492-wi-fi-modyl-nodemcu-esp8266](http://www.arduino.ua/prod1492-wi-fi-modyl-nodemcu-esp8266)
15. BME280 Combined humidity and pressure sensor [Электронный ресурс]: BST-BME280-DS001-18 – Режим доступа: [www / URL: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme280-ds002.pdf](http://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme280-ds002.pdf)

16. ESP8266 with BME280 using Arduino IDE (Pressure, Temperature, Humidity) [Електронний ресурс]. – Режим доступу: [www / URL: https://randomnerdtutorials.com/esp8266-bme280-arduino-ide/](http://www.randomnerdtutorials.com/esp8266-bme280-arduino-ide/)
17. RuslanNadyrshyn / diplom [Електронний ресурс]. – Режим доступу: [www / URL: https://github.com/RuslanNadyrshyn/diplom/tree/develop](https://github.com/RuslanNadyrshyn/diplom/tree/develop)
18. BME280 Nadyrshyn [Електронний ресурс]. – Режим доступу: [www / URL: https://weather.vps.webdock.cloud/](https://weather.vps.webdock.cloud/)
19. Український гідрометеорологічний центр Державної служби України з надзвичайних ситуацій [Електронний ресурс]. – Режим доступу: [www / URL: https://www.meteo.gov.ua/](https://www.meteo.gov.ua/)