

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ
ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА
«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»
Циклова комісія комп'ютерних систем та мереж
(повна назва циклової комісії)

Допустити до захисту
Голова випускової циклової комісії
комп'ютерних систем та мереж

(повна назва циклової комісії)

Ірина КРАВЧУК

(ім'я, ПРІЗВИЩЕ)

« 10 » 06 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬО-ПРОФЕСІЙНОГО СТУПЕНЯ
ФАХОВИЙ МОЛОДШИЙ БАКАЛАВР

Тема: Метеостанція з веб-інтерфейсом на базі ESP

Група: 3-013

Спеціальність: 123 «Комп'ютерна інженерія»

Здобувач освіти

Вячеслав КОЧЕТОВ
(підпис)

Вячеслав КОЧЕТОВ

(ім'я, ПРІЗВИЩЕ)

Керівник роботи

Тетяна РУБАН
(підпис)

Тетяна РУБАН

(ім'я, ПРІЗВИЩЕ)

Консультант з
оформлення
пояснювальної записки

Оксана ОСАДЧА
(підпис)

Оксана ОСАДЧА

(ім'я, ПРІЗВИЩЕ)

Кривий Ріг 2025 р.

КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ
ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА
«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»

Відділення комп'ютерної та програмної інженерії
Циклова комісія комп'ютерних систем та мереж
Освітньо-професійний ступінь фаховий молодший бакалавр
Спеціальність 123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Голова випускової циклової комісії
комп'ютерних систем та мереж

(повна назва циклової комісії)

Ірина КРАВЧУК

(ім'я, ПРІЗВИЩЕ)

(підпис)

« 01 » 03

2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ОСВІТИ

Кочетов Вячеслав Владиславович

(прізвище, ім'я, по батькові)

1. Тема роботи Метеостанція з веб-інтерфейсом на базі ESP

Керівник роботи Рубан Тетяна Миколаївна, викладач першої категорії

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по коледжу від « 04 » 04 2025 року № 50-ст

2. Строк подання здобувачем освіти роботи з _____ по _____

3. Вихідні дані до роботи Розробка портативного пристрою, який буде збирати інформацію про поточну температуру, вологість повітря, тиск, та візуалізовувати інформацію через веб-інтерфейс

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
Обґрунтування вибору напрямку, розробка автономного та портативного пристрою, створення функціональної схеми, програмування та створення веб-інтерфейсу, програмування мікроконтролера ESP8266

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Презентація Microsoft PowerPoint

6. Консультанти розділів роботи (проєкту)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Узгодження технічного завдання з керівником кваліфікаційної роботи	17.03.2025-21.03.2025	виконано
2	Підбір та вивчення науково-технічної літератури за темою кваліфікаційної роботи	24.03.2025-28.03.2025	виконано
3	Обґрунтування вибору програмних засобів	31.03.2025-04.04.2025	виконано
4	Опис компонентів. Обґрунтування їх вибору	07.04.2025-09.04.2025	виконано
5	Розробка програмного забезпечення	10.04.2025-28.04.2025	виконано
6	Дослідження ефективності реалізованих методів	29.04.2025-02.05.2025	виконано
7	Написання пояснювальної записки	12.05.2025-23.05.2025	виконано
8	Перевірка на плагіат пояснювальної записки	26.05.2025-30.05.2025	виконано
9	Попередній захист кваліфікаційної роботи	02.06.2025-06.06.2025	виконано
10	Захист кваліфікаційної роботи		

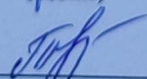
Здобувач освіти _____


(підпис)

Вячеслав КОЧЕТОВ _____

(ім'я, ПРІЗВИЩЕ)

Керівник роботи _____


(підпис)

Тетяна РУБАН _____

(ім'я, ПРІЗВИЩЕ)



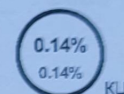
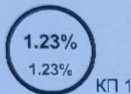
Звіт подібності

метадані

Назва організації
Ukrainian national aviation university
 Заголовок
КПІ_2025_123 Кочетов
 Автор Науковий керівник / Експерт
КочетовРубан Т.М
 підрозділ
Криворізький Фаховий коледж

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



25

Довжина фрази для коефіцієнта подібності 2

13071

Кількість слів

100305

Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		0
Інтервали		0
Мікропробіли		4
Білі знаки		0
Парафрази (SmartMarks)		13

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Копію тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	Копія тексту	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	http://dspace.wunu.edu.ua/bitstream/316497/51954/1/%D0%A8%D1%82%D0%B8%D0%BD%D0%B4%D0%B8.pdf		14 0.11 %
2	https://elartu.tntu.edu.ua/bitstream/1b/41874/2/Dyplom_Vorona_M_S_2023.pdf		12 0.09 %
3	2023_61220009_Pivtorak_Zakhar_Andriiovych_117677 11/21/2024 National University "Lviv Politechnika" (National University Lviv Politechnika)		11 0.08 %

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи з теми «Метеостанція з веб-інтерфейсом на базі *ESP*»: 64 сторінки основного тексту, 16 рисунків, 15 використаних джерел, 1 додаток.

ESP, МЕТЕОСТАНЦІЯ, МІКРОКОНТРОЛЕР, ПРОГРАМУВАННЯ, ДАТЧИК ТЕМПЕРАТУРИ, ДАТЧИК ТИСКУ, ДАТЧИК ВОЛОГОСТІ, ДАТЧИК РІВНЯ СВІТЛА, АВТОНОМНА СИСТЕМА, ВЕБ-ІНТЕРФЕЙС.

Метою кваліфікаційної роботи є розробка та програмування метеостанції з веб-інтерфейсом на базі мікроконтролера *ESP8266*. У ході виконання кваліфікаційної роботи було проведено моделювання, вибір апаратних компонентів та створення програмного забезпечення для автоматизованої системи збору та відображення метеорологічних даних. Програмна частина метеостанції реалізована за допомогою середовища *Arduino IDE* з використанням бібліотек для роботи з датчиками та протоколами обміну даними. Для відображення даних створено веб-інтерфейс, що дозволяє користувачеві у реальному часі переглядати інформацію про температуру, вологість, тиск, рівень освітленості та висоту над рівнем моря через *Wi-Fi*-з'єднання.

У третьому розділі кваліфікаційної роботи було розроблено структурну схему метеостанції, описано алгоритм роботи програми та визначено функціональне призначення кожного модуля системи.

В результаті дослідження було створено автоматизовану метеостанцію з веб-інтерфейсом, яка здатна збирати, обробляти та відображати метеорологічні дані в реальному часі. Отримані дані можна використовувати для подальшого аналізу, досліджень або інтеграції в більш складні автоматизовані системи моніторингу.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ.....	6
ВСТУП	7
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Основи параметри метеорологічних вимірювань.....	9
1.2 Огляд існуючих рішень метеостанцій	14
1.3 Аналіз технологій для збору та обробки метеоданих	21
РОЗДІЛ 2 ВИБІР АПАРАТНОЇ ТА ПРОГРАМНОЇ ЧАСТИНИ.....	27
2.1 Обґрунтування вибору мікроконтролера	27
2.2 Вибір сенсорів для вимірювання параметрів	30
2.3 Огляд технологій для створення веб-інтерфейсу	36
РОЗДІЛ 3 РОЗРОБКА МЕТЕОСТАНЦІЇ.....	42
3.1 Архітектура системи та схема підключення компонентів.....	42
3.2 Програмування мікроконтролера <i>ESP</i>	44
3.3 Збір та обробка метеоданих	50
3.4 Реалізація веб-інтерфейсу для візуалізації показників	54
3.5 Тестування та налагодження пристрою (інструкція користувача).....	56
ВИСНОВКИ.....	62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	63
ДОДАТОК А.....	65

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

I2C – *Inter-Integrated Circuit*, двопровідна шина для обміну даними між пристроями.

SPI – *Serial Peripheral Interface*, протокол послідовного зв'язку для передачі даних між мікроконтролером і периферійними пристроями

GPIO – *General Purpose Input/Output*, універсальні виводи вводу/виводу на мікроконтролері

MEMS – *Micro-Electro-Mechanical Systems*, мікроелектромеханічні системи, технологія, що використовується в датчиках.

ESP – Скорочення від *ESP8266*, мікроконтролер із вбудованим *Wi-Fi* модулем, що використовується в метеостанції

SON – *JavaScript Object Notation*, формат обміну даними між мікроконтролером і веб-інтерфейсом

REST API – *Representational State Transfer Application Programming Interface*, інтерфейс програмування для обміну даними

UART – *Universal Asynchronous Receiver/Transmitter*, інтерфейс для послідовного обміну даними

AJAX – *Asynchronous JavaScript and XML*, технологія асинхронного оновлення даних у веб-інтерфейсі

MQTT – *Message Queuing Telemetry Transport*, протокол для передачі даних у системах з обмеженими ресурсами

П'єзорезистивний сенсор – Датчик, який змінює опір під впливом механічної деформації

Фоторезистор – Датчик, що змінює опір залежно від інтенсивності світла

Адаптивний дизайн – Дизайн веб-інтерфейсу, що підлаштовується під розміри екрана пристрою

NodeMCU – Плата розробки на базі *ESP8266* для створення *IoT*-пристроїв

ВСТУП

Сучасний стан розвитку інформаційно-комунікаційних технологій, а також широке застосування мікроконтролерів та бездротових мереж створюють нові можливості для автоматизації збору та обробки даних у різних сферах діяльності людини. Однією з таких важливих сфер є метеорологія, яка має вирішальне значення для прогнозування погодних умов, моніторингу кліматичних змін та забезпечення безпеки у багатьох галузях, таких як сільське господарство, енергетика, транспорт, туризм, будівництво та охорона здоров'я. Особливої актуальності набувають системи моніторингу навколишнього середовища, які дозволяють у режимі реального часу відстежувати зміни атмосферних параметрів та своєчасно реагувати на потенційні загрози. Завдяки стрімкому розвитку технологій Інтернету речей (IoT) з'явилася можливість створювати доступні, енергоефективні та мобільні пристрої для збору метеоданих, які здатні інтегруватися з веб-інтерфейсами та хмарними сервісами.

Метою даної кваліфікаційної роботи є розробка метеостанції з веб-інтерфейсом на базі мікроконтролера ESP8266, яка забезпечить збір та обробку основних метеорологічних параметрів — температури повітря, атмосферного тиску, рівня освітленості — та надаватиме користувачам доступ до даних у режимі реального часу через веб-браузер. Такий підхід дозволяє значно розширити функціональні можливості традиційних метеостанцій, а також створює передумови для віддаленого моніторингу та інтеграції з іншими автоматизованими системами.

Для реалізації цієї мети у роботі проведено аналіз існуючих рішень метеостанцій та обґрунтовано вибір апаратної і програмної частини системи. Зокрема, розглянуто переваги використання мікроконтролера ESP8266, який має вбудований модуль Wi-Fi, компактні розміри, низьке енергоспоживання та підтримку різноманітних сенсорів. Це робить його ідеальним вибором для створення сучасної метеостанції з можливістю віддаленого доступу.

Актуальність теми полягає в тому, що запропонована система здатна працювати автономно, з мінімальними витратами ресурсів, і водночас забезпечує користувача достовірною інформацією про стан навколишнього середовища. Така система може використовуватися як у приватних домоволодіннях, так і на підприємствах, у сільському господарстві, для наукових досліджень або навіть у навчальних цілях.

Таким чином, кваліфікаційна робота спрямована на розробку доступної, надійної та ефективної метеостанції, яка поєднує апаратну частину на основі сучасного мікроконтролера та зручний веб-інтерфейс, що відповідає вимогам часу та сучасним стандартам у сфері інформаційних технологій.

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Основі параметри метеорологічних вимірювань

Метеорологічні вимірювання є однією з найважливіших складових системи моніторингу стану навколишнього середовища, яка дає змогу отримати об'єктивну інформацію про стан атмосфери, що оточує нашу планету Земля. Ці вимірювання є основою для всіх прогнозів погоди, досліджень клімату та багатьох інших видів діяльності, які прямо або опосередковано залежать від погодних умов. Без наявності якісних і регулярних метеорологічних вимірювань неможливо було б розуміти та аналізувати природні явища, їхню мінливість та динаміку розвитку, а отже, й вчасно реагувати на потенційні загрози, які виникають унаслідок зміни клімату або стихійних лих. Такі вимірювання дозволяють нам краще розуміти складні процеси, що відбуваються в атмосфері, і на основі цього прогнозувати можливі наслідки для різних сфер людської діяльності, таких як сільське господарство, енергетика, транспорт, будівництво, туризм, охорона здоров'я та багато інших. Усі ці галузі економіки та соціальної сфери тією чи іншою мірою залежать від погоди: наприклад, фермери повинні знати про можливі заморозки або посухи, щоб запобігти втратам урожаю, енергетики — про екстремальні температури, які впливають на споживання електроенергії, транспортники — про умови руху транспорту в разі сильного вітру або опадів, а будівельники — про можливість проведення будівельних робіт під час несприятливих погодних умов. Основною метою метеорологічних вимірювань є безперервна та максимально точна фіксація параметрів навколишнього середовища, що впливають на життя людей, на функціонування інфраструктури, на стан природних екосистем та на безліч інших аспектів нашого повсякденного життя. До таких параметрів належать температура повітря, відносна та абсолютна вологість, атмосферний тиск, швидкість і напрямок вітру, кількість та інтенсивність опадів, а також цілий

ряд додаткових величин, серед яких можна виокремити ультрафіолетове випромінювання, сонячну радіацію, температуру та вологість ґрунту, електричну активність атмосфери, вміст пилу, концентрацію газів (озону, вуглекислого газу, метану) та багато інших, які стали доступними для вимірювань завдяки розвитку технологій та наукових досягнень. Кожен із цих параметрів має своє власне, надзвичайно важливе практичне значення, фізичні особливості, характерні закономірності зміни у просторі та часі, а також спеціальні методи вимірювання, які постійно вдосконалюються й оновлюються в міру розвитку науки і техніки. Сучасні метеорологічні вимірювання вже не обмежуються використанням простих механічних приладів, як це було на початку розвитку цієї науки, а широко застосовують складні електронні системи з високою точністю та автоматизацією процесу, що дозволяє забезпечити більш стабільні та тривалі вимірювання. температура повітря є одним із найосновніших і найбільш поширених метеорологічних параметрів, що характеризує ступінь нагрівання або охолодження повітря у конкретному місці та у визначений момент часу. Вона визначає багато атмосферних процесів, серед яких: утворення хмар, виникнення опадів, рух повітряних мас, процеси випаровування, теплообмін між поверхнею землі та атмосферою, що, своєю чергою, впливає на погодні умови та формування клімату певного регіону. Температура має вирішальний вплив на здоров'я та самопочуття людей, на життєдіяльність рослин і тварин, на ріст і розвиток сільськогосподарських культур, а також на працездатність машин і обладнання. Для вимірювання температури повітря використовуються різні типи термометрів. Спочатку це були ртутні та спиртові пристрої, які давали можливість вимірювати температуру з достатньою точністю, проте потребували постійного нагляду та обслуговування. З розвитком технологій ці прилади поступово замінили термоелектричні, терморезистивні та інші електронні датчики, які не тільки підвищили точність вимірювань, а й забезпечили можливість автоматизованого збору даних, передачі їх на

відстань і тривалого використання без необхідності регулярного ручного обслуговування.

Щоб уникнути впливу прямих сонячних променів, опадів та інших факторів, які можуть спотворити результати вимірювань, термометри встановлюють у спеціальні захисні кожухи або радіаційні екрани, що створюють умови для вільної циркуляції повітря навколо вимірювального елемента та забезпечують об'єктивність отриманих даних. Варто зазначити, що значення температури можуть суттєво змінюватися залежно від пори року, часу доби, географічної широти, висоти над рівнем моря, наявності водойм, лісових масивів або інших природних та антропогенних факторів, які впливають на мікроклімат місцевості.

Вологість повітря — ще один ключовий параметр, який характеризує вміст водяної пари в атмосфері та має велике значення для багатьох природних і техногенних процесів. Вологість визначає комфортність умов для людини, впливає на швидкість випаровування води з поверхонь та на процеси конденсації, які, своєю чергою, впливають на утворення хмар, туманів, опадів та загальну динаміку атмосфери. Вологість буває абсолютною — коли визначають масу водяної пари в одному кубічному метрі повітря, і відносною — коли визначають, наскільки повітря насичене вологою при певній температурі, що виражається у відсотках. Раніше для вимірювання вологості використовували волосні або конденсаційні гігрометри, які мали свої обмеження щодо точності та стабільності показників. Однак у сучасних системах усе частіше використовують ємнісні, електронні та резистивні датчики, які відзначаються високою точністю, стабільністю та можливістю автоматизації процесу вимірювання, що значно полегшує моніторинг стану атмосфери та підвищує надійність зібраних даних. Варто зауважити, що параметри вологості змінюються протягом доби, залежать від температури повітря, наявності водойм, випадіння опадів та сезонних особливостей клімату регіону.

Атмосферний тиск — це сила, з якою стовп повітря тисне на земну поверхню, і він є одним із головних показників, що визначає загальний стан атмосфери та її поведінку. Зміни атмосферного тиску можуть сигналізувати про наближення певних метеорологічних систем, таких як циклони чи антициклони, а також бути передвісниками змін погоди, наприклад, підвищення ймовірності опадів, посилення вітру або зменшення хмарності. Як правило, високий атмосферний тиск асоціюється з ясною та стабільною погодою, тоді як зниження тиску часто свідчить про можливе погіршення погодних умов. Для вимірювання атмосферного тиску використовують різні типи барометрів: ртутні, анероїдні та електронні. У сучасних системах найчастіше застосовують електронні барометри, які працюють на основі п'єзоелектричного або ємнісного принципу. Вони дають змогу перетворювати механічні зміни на електричний сигнал і таким чином інтегрувати дані в сучасні автоматизовані системи збору та обробки метеорологічної інформації. Варто враховувати, що значення атмосферного тиску змінюється з висотою над рівнем моря, тому для можливості порівняння даних з різних регіонів світу та для уніфікації інформації використовують перерахунок до умов рівня моря, що дозволяє зіставляти дані з різних висот і робити об'єктивні висновки.

Швидкість і напрямок вітру — це ще два важливих параметри, які характеризують стан атмосфери і мають велике практичне значення для багатьох галузей діяльності людини. Вітер виникає внаслідок різниці атмосферного тиску між різними регіонами, а його напрямок вказує, звідки рухається повітря, що важливо для прогнозування погоди, аналізу транспортних потоків забруднювальних речовин, розрахунків вітрових навантажень у будівництві, організації польотів літальних апаратів, морських подорожей та інших видів діяльності. Вітер також має значення для енергетики, оскільки є джерелом відновлюваної енергії, яка може використовуватися для виробництва електроенергії. Для вимірювання швидкості вітру використовуються анемометри, які можуть бути механічними (наприклад, чашкові або пропелерні) або безконтактними (ультразвукові,

лазерні). Сучасні ультразвукові анемометри відзначаються високою точністю та відсутністю рухомих частин, що робить їх більш довговічними та зручними в обслуговуванні. Напрямок вітру вимірюється флюгерами, які за допомогою рухомої стрілки показують сторону, з якої дмухає вітер.

У багатьох сучасних метеорологічних станціях ці два прилади поєднують в єдиний вимірювальний вузол для зручності обробки даних та їхнього автоматизованого збору. Кількість опадів є важливим показником, який характеризує сумарну кількість води, що випала на певну площу за певний проміжок часу. Оподи можуть бути різними за формою та агрегатним станом: дощ, сніг, град, мряка, роса. Цей показник є критичним для оцінки водного балансу, планування зрошувальних робіт у сільському господарстві, прогнозування можливих паводків, контролю за зволоженням ґрунтів та багатьох інших завдань, пов'язаних із використанням водних ресурсів.

Раніше для вимірювання кількості опадів використовували прості осадкоміри, які представляли собою відкриті посудини з градуйованою шкалою, що дозволяло визначати кількість води, яка випала. Проте такі пристрої вимагали ручного зчитування даних і регулярного обслуговування. З розвитком технологій ці прилади були замінені автоматичними осадкомірами, що працюють за принципом зважування води або оптичної фіксації крапель, які перетинають світловий промінь, що дозволяє отримати більш точні та надійні дані.

Однією з особливостей вимірювання опадів є зміна агрегатного стану води: наприклад, сніг накопичується у вигляді снігової шапки, яку необхідно або розтоплювати, або використовувати спеціальні обігрівальні елементи для того, щоб отримати точні дані про кількість опадів. Крім сумарної кількості опадів, важливим є також показник інтенсивності опадів, який визначає швидкість їх випадання та дозволяє оцінити потенційні ризики паводків або підтоплень.

Окрім основних параметрів, у метеорології все частіше фіксують додаткові величини, такі як рівень ультрафіолетового випромінювання,

сонячну інсоляцію, температуру та вологість ґрунту, електричну активність атмосфери, кількість пилю, концентрацію газів (озону, вуглекислого газу, метану) та багато інших. Завдяки розвитку технологій і використанню сучасних сенсорів стало можливим проводити комплексні дослідження стану атмосфери, що дозволяє створювати більш повну картину стану довкілля.

Отже, метеорологічні вимірювання — це складний, багатокомпонентний процес, який потребує максимальної точності, регулярності та надійності. Саме завдяки таким вимірюванням формується наукова база для вивчення клімату, для прогнозування погодних умов, для прийняття стратегічно важливих рішень на національному та глобальному рівнях, а також для забезпечення безпеки й добробуту всіх людей на нашій планеті.

1.2 Огляд існуючих рішень метеостанцій

На сучасному ринку метеорологічного обладнання представлений надзвичайно широкий спектр рішень, які суттєво відрізняються між собою за багатьма параметрами, серед яких варто відзначити функціональність, технічні характеристики, точність проведення вимірювань, складність конструкції, тип живлення, методи передачі даних, а також, безумовно, вартість. Такий великий і різноманітний вибір дозволяє кожному користувачеві підібрати метеостанцію відповідно до власних потреб і можливостей — починаючи від найпростіших моделей, призначених переважно для особистого або побутового використання, і завершуючи складними та дорогими професійними системами, які здебільшого застосовуються в наукових дослідженнях, на спеціалізованих метеорологічних станціях, в аеропортах для забезпечення авіаційної безпеки або ж в аграрному секторі для максимально точного моніторингу кліматичних умов з метою підвищення ефективності сільськогосподарського виробництва. Метеостанції можна умовно поділити на кілька основних категорій, серед яких

варто виділити побутові, напівпрофесійні та професійні моделі. Кожна з цих категорій має свої характерні особливості щодо функціональних можливостей, ступеня точності вимірювань, способу інсталяції та монтажу, тривалості експлуатації, а також можливості інтеграції з іншими системами або мережами передачі даних.

Одним із важливих критеріїв вибору для потенційного покупця, безумовно, є ціна пристрою, яка значною мірою визначається усіма вищезгаданими характеристиками, а також брендом виробника та рівнем якості виготовлення. Для початку варто більш детально зупинитися на побутових метеостанціях, оскільки саме вони є найбільш поширеним типом пристроїв, орієнтованих переважно на використання звичайними користувачами в умовах власного будинку або квартири. Побутові метеостанції зазвичай відзначаються компактними розмірами, простотою монтажу та надзвичайною зручністю у використанні навіть для людей без спеціальної технічної підготовки. Основною метою побутових моделей є надання користувачеві базової інформації про погодні умови в реальному часі, що дає змогу планувати повсякденні справи з урахуванням актуальних кліматичних умов. Здебільшого такі прилади можуть здійснювати вимірювання температури повітря (як зовнішньої, так і внутрішньої), рівня вологості, атмосферного тиску, а також нерідко оснащуються додатковою функцією прогнозу погоди, яка ґрунтується на зміні барометричного тиску впродовж певного проміжку часу. Для відображення вимірюваної інформації побутові метеостанції зазвичай мають невеликий вбудований дисплей, який може бути як монохромним, так і кольоровим, залежно від конкретної моделі та виробника. На такому дисплеї зазвичай відображаються значення температури, вологості, час, дата, а також індикатор тенденції зміни погодних умов, що може бути реалізований у вигляді різноманітних іконок, наприклад із зображеннями хмарності або сонця. Окрім цього, деякі моделі додатково мають функції будильника, календаря, відображення фаз місяця, а іноді навіть підтримують можливість підключення зовнішніх бездротових датчиків, що

дозволяє користувачу контролювати параметри одразу в кількох точках — наприклад, як у будинку, так і на вулиці. До основних переваг побутових метеостанцій можна віднести їхню відносно невисоку ціну, простоту експлуатації, широку доступність у магазинах електроніки або інтернет-магазинах, а також мінімальні вимоги до технічних знань і навичок користувача. Разом з тим необхідно зазначити, що точність вимірювань у таких приладах часто є нижчою порівняно з професійними моделями, а термін служби безпосередньо залежить від умов експлуатації, зокрема від рівня вологості, температурних перепадів та якості захисту датчиків від несприятливих зовнішніх впливів. У деяких моделях побутових метеостанцій також передбачена базова функція прогнозу погоди, яка працює на основі змін атмосферного тиску, зафіксованих упродовж певного періоду часу. Такий прогноз зазвичай дає змогу користувачу скласти загальне уявлення про ймовірність змін погодних умов у найближчі години чи добу, хоча й не гарантує максимальної точності, притаманної професійним системам моніторингу. Проте для більшості побутових користувачів цього цілком достатньо для планування своїх щоденних справ та активностей. Серед найвідоміших і популярних представників побутових метеостанцій варто згадати такі моделі, як *Oregon Scientific BAR208HG*, *La Crosse Technology WS6812*, а також *AcuRite 01512*. Кожна з цих моделей має власний набір функцій, індивідуальний стильний дизайн і певною мірою відрізняється способом індикації вимірюваних параметрів. Наприклад, модель *BAR208HG* від *Oregon Scientific* вирізняється своїм інформативним дисплеєм, на якому відображаються графічні символи, що полегшують сприйняття даних. Окрім цього, ця модель має функцію прогнозу погоди, можливість вимірювання температури зовнішнього середовища через бездротовий датчик і оснащена індикаторами комфортності, що дозволяють швидко оцінити поточні умови. Модель *WS6812* від *La Crosse Technology* також підтримує можливість підключення зовнішнього датчика температури, має вбудований годинник з

радіоконтролем, що автоматично синхронізує час, а її елегантний дизайн робить її універсальною для використання в інтер'єрі будь-якого приміщення.

Ще одним прикладом є модель *AcuRite 01512*, яка вирізняється дещо ширшими можливостями порівняно з іншими побутовими моделями: вона дозволяє вимірювати не лише температуру та вологість, а й напрямок та швидкість вітру, кількість опадів, а також має великий кольоровий дисплей, що робить зручним перегляд даних у будь-яких умовах освітлення. До того ж ця модель підтримує бездротову передачу даних, що значно спрощує інтеграцію з іншими пристроями. Деякі модифікації цієї метеостанції також мають функцію збереження історичних даних вимірювань та синхронізацію з мобільними додатками або інтернет-сервісами, що відкриває перед користувачем додаткові можливості для аналізу погодних тенденцій упродовж певного часу. На рисунку 1.1 зображено одну з таких побутових метеостанцій, що ілюструє їхній зовнішній вигляд, компоновання дисплея та розміщення основних показників.



Рисунок 1.1 – Побутова метеостанцій *Oregon Scientific*

Такі метеостанції є досить зручними для використання та доступними за своєю вартістю, особливо для тих людей, які не володіють спеціальними знаннями в галузі метеорології. Однак, попри їхню зручність та відносну дешевизну, вони дещо поступаються професійним метеостанціям як за точністю вимірювань, так і за строком служби. Саме тому для реалізації даної

кваліфікаційної роботи, зокрема для інтеграції з веб-інтерфейсом, такий тип метеостанцій, на жаль, не зовсім підходить.

Що стосується професійних метеостанцій, то вони призначені передусім для використання у спеціалізованих метеорологічних службах, а також в аеропортах будь-якого типу чи у сільському господарстві. Подібні системи зазвичай оснащуються різноманітними сенсорами, які дозволяють вимірювати температуру повітря, вологість, атмосферний тиск, швидкість та напрямок вітру, кількість опадів, а також рівень сонячного випромінювання. Прикладами таких моделей можуть бути *Vaisala WXT536*, *Campbell Scientific CS215* та *Davis Instruments Vantage Pro2*. Вони характеризуються високою точністю отриманих даних, надійною конструкцією та здатністю безвідмовно функціонувати навіть в умовах екстремального клімату. На рисунку 1.2 представлено одну з професійних метеостанцій.



Рисунок 1.2 – Метеостанція *Davis Instruments Vantage Pro2*

Крім того, такі метеостанції зазвичай оснащуються системами передачі даних різними способами, зокрема через кабельне або бездротове з'єднання, що дає їм можливість працювати з веб-інтерфейсами. Разом із тим, досить висока ціна та складність інсталяції можуть стати певною перешкодою для їх масового використання. При розробці власної метеостанції доцільно брати за основу ті переваги, якими володіють професійні системи, зокрема їхню точність вимірювань та можливість модульного розширення, і намагатися

перенести ці характеристики у більш доступний і бюджетний формат. Останнім часом все більшої популярності набули бездротові метеостанції, що пояснюється зручністю встановлення та гнучкістю у виборі місця для розташування датчиків. Такі станції передають дані з віддалених сенсорів на основну станцію без використання кабелів, що є особливо корисним для встановлення на великій площі або у важкодоступних місцях. До прикладів таких пристроїв можна віднести *Ambient Weather WS-2902C*, *Netatmo Weather Station* та *Froggit WH3000*. Бездротові станції зазвичай використовують радіозв'язок, наприклад *Wi-Fi* або *Bluetooth*, для передачі даних, що робить їх особливо зручними для інтеграції з веб-інтерфейсами. Такі системи досить прості в експлуатації і часто вирізняються стильним дизайном, що робить їх привабливими для кінцевих користувачів.

Водночас бездротові технології можуть мати і свої недоліки, зокрема проблеми з радіоперешкодами або обмежений радіус дії, що вимагає ретельного тестування таких рішень перед впровадженням. Для метеостанції, яка повинна працювати з веб-інтерфейсом, наявність бездротової передачі даних є важливою функцією, адже саме вона дозволяє реалізувати можливість віддаленого доступу. Інтернет-метеостанції відносяться до порівняно нової категорії пристроїв, яка активно розвивається завдяки широкому впровадженню хмарних технологій. Такі системи не лише збирають необхідні метеодані, а й передають їх у мережу Інтернет, що дає змогу користувачам у реальному часі відслідковувати погодні умови в будь-якій точці світу. Прикладами таких систем є *Weather Underground Personal Weather Station*, *Ecovitt GW1000* та *Tempest Weather System*. Ці пристрої часто підтримують

інтеграцію з онлайн-платформами, де користувачі можуть ділитися зібраними даними, проводити їх аналіз і порівнювати з показниками з інших станцій.

Такий підхід є особливо актуальним для метеостанцій з веб-інтерфейсом, оскільки він забезпечує доступність даних та зручність їх використання.

Проте для функціонування таких систем необхідне стабільне підключення до Інтернету, а також відповідний рівень захисту інформації, що є важливим аспектом при створенні власної системи. На рисунку 1.3 зображено модель одної з таких метеостанцій.



Рисунок 1.3 – Інтернет-метеостанції *Ecowitt GW1000*

Мобільні метеостанції розроблені спеціально для роботи у виїзних умовах, де потрібна оперативність і мобільність. Вони є незамінними під час дослідницьких експедицій, у сільському господарстві, на будівельних об'єктах, а також у ситуаціях, де стаціонарне обладнання встановити неможливо або недоцільно. Завдяки своїй компактності та невеликій вазі, такі прилади легко транспортувати, встановлювати та обслуговувати. Серед відомих прикладів подібних пристроїв можна назвати *Kestrel 5500 Weather Meter*, *Skywatch BL500* і *WeatherFlow WEATHERmeter*, які завдяки своїй простоті та ергономічності дозволяють оперативно збирати найнеобхідніші дані про температуру, рівень вологості повітря, а також швидкість та напрямок вітру. Такі пристрої, як правило, мають можливість синхронізації з мобільними додатками через *Bluetooth*, що значно полегшує збір даних та подальший їх аналіз. Основними перевагами таких мобільних рішень є їхня

універсальність і здатність працювати в різних кліматичних умовах, а також автономність, яка забезпечується завдяки живленню від акумуляторів або батарейок. Водночас варто зазначити, що функціонал таких станцій обмежується лише базовими метеорологічними параметрами, без можливості вимірювати атмосферний тиск, кількість опадів або рівень сонячної радіації. Це дещо звужує сферу їх застосування у порівнянні з більш складними системами, які використовуються у метеорологічних службах або наукових дослідженнях. Через відсутність прямого підключення до мережі Інтернет та обмежений набір функцій мобільні метеостанції рідко використовуються для тривалого моніторингу або для збору великих обсягів даних, які потребують постійного зберігання та обробки на сервері. Однак їхню компактність і портативність варто враховувати як додаткову опцію при розробці власної метеостанції, що могла б працювати не лише стаціонарно, а й у виїзному режимі. Це відкриває можливість створення гібридного рішення, яке об'єднуватиме найкращі якості як професійних, так і мобільних систем. Отже, серед широкого спектра доступних рішень – від найпростіших портативних моделей до складних багатофункціональних комплексів – можна виділити найкорисніші характеристики та інтегрувати їх у власну розробку.

Завдяки цьому вдасться створити пристрій, що не лише забезпечить високу точність вимірювань, але й дозволить працювати в екстремальних умовах та передавати дані в реальному часі через веб-інтерфейс, тим самим роблячи систему зручною та універсальною для користувачів із різними потребами.

1.3 Аналіз технологій для збору та обробки метеоданих

Аналогові датчики, які застосовуються у метеорологічних системах для вимірювання різноманітних фізичних величин, таких як температура, атмосферний тиск, вологість повітря, швидкість та напрямок вітру, являють собою традиційний спосіб отримання необхідних даних про стан

навколишнього середовища. Ці сенсори працюють за принципом перетворення фізичних характеристик середовища у відповідний аналоговий сигнал, який відображається у вигляді зміни механічних або електричних параметрів. Наприклад, класичні ртутні або спиртові термометри визначають температуру шляхом зміни висоти стовпчика рідини у спеціальній капілярній трубці, а анероїдні барометри визначають зміну атмосферного тиску за допомогою деформації гнучкої металевої мембрани, яка реагує на зміни тиску повітря. Подібним чином, механічні гігromетри визначають рівень вологості через натяг чи послаблення волоскових або органічних елементів. Основною перевагою таких аналогових сенсорів є їхня відносна простота у використанні, низька вартість виробництва та експлуатації, а також відсутність потреби у додатковому джерелі живлення у багатьох випадках, що особливо важливо для віддалених районів та місць, де відсутня електромережа. Це робить їх дуже корисними для базових завдань моніторингу навколишнього середовища у польових умовах або під час проведення досліджень у природних умовах.

Однак, слід зазначити, що механічні обмеження конструкції аналогових сенсорів можуть негативно впливати на точність та стабільність вимірювань, особливо при тривалому використанні або під час зміни погодних умов, коли температура чи вологість можуть спричинити зміни в матеріалах самого сенсора. Крім того, аналогові сенсори потребують ручного зчитування або використання додаткових систем для перетворення аналогового сигналу у цифровий, що значно ускладнює їх інтеграцію у сучасні автоматизовані системи збору та обробки інформації. Саме тому для розробки та реалізації веб-інтерфейсу, де дані повинні передаватися у цифровому форматі для обробки та відображення, аналогові сенсори не є найкращим вибором, оскільки вони вимагають використання додаткових схем перетворення та складних процедур калібрування. На відміну від них, цифрові сенсори останніми роками набули широкого поширення завдяки своїм численним перевагам у точності, швидкості передачі даних та сумісності із сучасними мікроконтролерами та комп'ютерами. Такі сенсори безпосередньо

конвертують фізичні параметри навколишнього середовища у цифровий код, що значно спрощує процес обробки інформації та дозволяє будувати автоматизовані системи збору даних у режимі реального часу. Наприклад, цифрові терморезистори або термопари змінюють свій електричний опір залежно від температури, а спеціалізовані ємнісні гігromетри вимірюють рівень вологості через зміну електричної ємності між пластинами сенсора, що дозволяє досягати високої точності та стабільності вимірювань протягом тривалого часу. Також існують ультразвукові анемометри, які визначають швидкість і напрямок вітру на основі часу проходження ультразвукових хвиль між передавачем та приймачем, що забезпечує точні дані навіть за складних погодних умов. Цифрові сенсори, як правило, мають високу стабільність та низьку похибку, що є особливо важливим для тривалого та безперервного моніторингу. Їх можна легко інтегрувати у сучасні мікроконтролери, такі як *Arduino*, *ESP32* чи *Raspberry Pi*, що забезпечує універсальність застосування та можливість налаштування системи під специфічні потреби користувача. Проте такі сенсори потребують зовнішнього джерела живлення, що може бути певним недоліком для автономних чи мобільних проєктів, де використання батарей або сонячних панелей має обмежений ресурс роботи. Також цифрові сенсори зазвичай мають вищу вартість у порівнянні з аналоговими, що потрібно враховувати під час проєктування та реалізації проєкту у випадку обмеженого бюджету. Незважаючи на зазначені недоліки, для системи збору метеорологічних даних із веб-інтерфейсом саме цифрові сенсори залишаються найкращим вибором, оскільки вони забезпечують повну сумісність із сучасними програмними платформами, дозволяють швидко отримувати дані та інтегрувати їх у хмарні сервіси. Застосування бездротових технологій, таких як *Wi-Fi*, *Bluetooth*, *Zigbee*, *GSM* та *LTE*, дозволяє передавати дані від сенсорів до центральної бази даних або безпосередньо у хмару, що значно спрощує

встановлення метеостанції та розширює її функціональні можливості. Один з прикладів таких сенсорів буде зображений на рисунку 1.4.



Рисунок 1.4 – Датчик *SHT35*

Це дає можливість розташовувати сенсори у важкодоступних або небезпечних місцях, мінімізуючи потребу у прокладанні фізичних кабелів та додаткової інфраструктури, а також забезпечує віддалений моніторинг стану довкілля через інтернет. Наприклад, за допомогою модуля *Wi-Fi* можна підключити метеостанцію до домашньої мережі або локального сервера, забезпечуючи постійний обмін даними, тоді як *Bluetooth* використовується для короткодистанційного зв'язку між сенсором та мобільним додатком для зчитування та відображення даних безпосередньо на смартфоні. Однак бездротові технології мають певні обмеження, такі як залежність від батарейного живлення, можливі перешкоди у передачі сигналу через фізичні об'єкти або інші електронні пристрої, а також обмежений радіус дії, що потребує ретельного планування мережевої інфраструктури та вибору оптимальних технологій під конкретні завдання. Хмарні сервіси відкривають додаткові можливості для централізованого зберігання, обробки та візуалізації метеорологічних даних, адже дані, передані від сенсорів, зберігаються на віддалених серверах, де вони можуть бути проаналізовані, агреговані та представлені у зручному для користувача форматі через веб-інтерфейс або мобільний додаток. Такі сервіси, як *Weather Underground* або *Ecowitt Cloud*, надають можливість відстежувати зміни кліматичних параметрів у реальному часі, створювати прогнози, порівнювати дані з іншими станціями та формувати статистичні звіти, що особливо актуально для сучасних проєктів у

галузі моніторингу довкілля. Приклад такого сервісу можна побачити на рисунку 1.5.



Рисунок 1.5 – Сервіс *Ecowitt Cloud*

Проте впровадження хмарних технологій вимагає стабільного та швидкого інтернет-з'єднання, а також відповідного рівня захисту даних від несанкціонованого доступу, що передбачає впровадження систем шифрування та автентифікації. Мікроконтролери та одноплатні комп'ютери виконують ключову роль у процесі збору, попередньої обробки та передачі даних, адже саме вони забезпечують взаємодію між сенсорами та кінцевим користувачем. *Arduino*, завдяки своїй простоті та доступності, підходить для реалізації базових функцій, таких як вимірювання температури, вологості та тиску, а також передачі даних через дротові або бездротові інтерфейси. *Raspberry Pi* відрізняється високою обчислювальною потужністю та універсальністю, що дозволяє реалізувати локальний веб-сервер, інтегрувати додаткові модулі, такі як дисплеї, камери, датчики якості повітря, та забезпечити роботу з базами даних для довготривалого зберігання та обробки інформації. *ESP32* поєднує в собі компактність, наявність вбудованого *Wi-Fi* та *Bluetooth*, а також багатозадачну архітектуру, що робить його ідеальним вибором для створення автономних метеостанцій, здатних працювати тривалий час без підключення до стаціонарної електромережі. Для повноцінної реалізації системи збору та обробки даних важливим є програмне забезпечення, яке відповідає за

зберігання, обробку, візуалізацію та передачу метеорологічної інформації у зручному для користувача форматі.

Для цього можна використовувати як універсальні мови програмування, наприклад, *Python* із бібліотеками *Pandas*, *NumPy*, *Matplotlib*, які дозволяють проводити гнучкий аналіз та створювати інтерактивні графіки, так і спеціалізовані програмні платформи, такі як *WeatherLink* або *Cumulus*, які забезпечують швидкий запуск та базову функціональність із мінімальними витратами часу. *Python* також дозволяє створювати веб-додатки за допомогою таких фреймворків, як *Flask* або *Django*, що забезпечує масштабованість, адаптивність та можливість реалізації унікальних рішень для конкретного проєкту. Таким чином, сучасна метеостанція з веб-інтерфейсом найкраще реалізується із застосуванням цифрових сенсорів, бездротових технологій передачі даних, хмарних сервісів та програмованих контролерів, що забезпечують високу точність вимірювань, автоматизацію процесів збору та обробки інформації, а також зручність доступу до даних у режимі реального часу з будь-якої точки світу.

РОЗДІЛ 2

ВИБІР АПАРАТНОЇ ТА ПРОГРАМНОЇ ЧАСТИНИ

2.2 Обґрунтування вибору мікроконтролера

Мікроконтролер *ESP8266* є одним із найпоширеніших і найефективніших рішень для реалізації проектів у сфері Інтернету речей, зокрема для створення метеостанцій із веб-інтерфейсом, завдяки своїм унікальним характеристикам і широким можливостям, які відповідають потребам такого роду розробок. Його вибір для подібного проекту є логічним і виправданим з огляду на численні технічні та практичні переваги, які забезпечують не лише високу функціональність, а й зручність у процесі створення системи. Цей мікроконтролер вирізняється своєю здатністю поєднувати в собі простоту використання, доступність і достатню продуктивність, що робить його привабливим як для досвідчених інженерів, так і для новачків у сфері електроніки та програмування.

Одна з ключових особливостей *ESP8266* – це вбудований *Wi-Fi* модуль, який дозволяє пристрою, а в нашому випадку метеостанції підключатися до бездротової мережі, без необхідності в окремому модулі зв'язку. Завдяки цьому, мікроконтролер може передавати зібрану метеорологічну інформацію у реальному часі, даючи доступ користувачам до метеоданих у реальному часі.

Також *ESP8266* підтримує протоколи *TCP/IP*, що достатньо сильно спрощує взаємодію пристрою з веб-серверами. Ще однією важливою характеристикою мікроконтролера є його енергоефективність, вона має значну роль у створенні метеостанції, адже вона повинна працювати автономно протягом тривалого часу. *ESP* оснащений кількома режимами роботи, одна з яких це режим глибокого сну, що дозволяє суттєво знизити енергоспоживання, коли метеостанція не виконує активних завдань. Сам мікроконтролер зображено на рисунку 2.1.



Рисунок 2.1 – Мікроконтролер *ESP8266*

Тобто метеостанція буде періодично прокидатись за допомогою *ESP8266* для зчитування даних з датчиків, та відправляти їх через *Wi-Fi* модуль, а вже потім повертатися у сон, це сильно економить заряд акумулятора. Наступною важливою характеристикою – є компактність, яка надає мікроконтролеру значних переваг при створення метеостанції. Через невеликі розміри полегшується інтегрування в обмежений простір корпусу, де будуть тісно знаходитися разом з мікроконтролером, датчики та живлення. Невелика вага такого модуля сприяє тому, що метеостанція залишається портативною, що є суттєвим плюсом для системи, яка буде переноситися з місця на місце або встановлюватися в польових умовах без значних зусиль. Обчислювальна потужність мікроконтролера забезпечується 32-бітним процесором *Tensilica L106* з тактовою частотою у 80 МГц, яку за потреби можна розігнати до 160 МГц, це дозволяє мікроконтролеру продуктивно справлятися з обробкою масиву даних, які будуть отримуватися з метеостанції.

Такої ефективності буде достатньо для того, щоб виконувати всі необхідні обчислення, такі як: температура, атмосферний тиск, рівень вологості повітря, рівень освітленості. Також, цієї потужності буде достатньо для запуску локального сервера, який буде забезпечувати відображення даних через веб-інтерфейс. Обсяг пам'яті мікроконтролера складається з 64 КБ для інструкцій і 96 КБ для даних, це дозволяє створювати складні програми, для одночасного збору інформації, обробки її та подальше надавання інформації

користувачу у зручному доступі через веб-браузер. Ця самодостатність усуває проблему у підключенні додаткових обчислювальних пристроїв.

Гнучкість *ESP* проявляється в його здатності працювати з широким спектром датчиків та периферійних пристроїв, що є важливим аспектом для метеостанції з різноманітними вимірювальними функціями. Гнучкість *ESP8266* розкривається в його здатності взаємо працювати з великим спектром датчиків та периферійних пристроїв, що надзвичайно необхідно для метеостанцій із різними вимірювальними можливостями. Мікроконтролер обладнаний декількома пінами *GPIO*, які дають змогу під'єднувати як аналогові, так і цифрові датчики, приміром, *DHT22* для вимірювання температури й вологості або *BMP280* для визначення атмосферного тиску.

Підтримка інтерфейсів *I2C*, *SPI* та *UART* значно розширює можливості інтеграції з більш складними модулями, зокрема датчиками швидкості вітру або опадів, що робить *ESP8266* універсальним рішенням для конструювання метеостанції із широким набором параметрів. Доступність і низька вартість *ESP8266* є ще одним вагомим аргументом на користь його використання в кваліфікаційній роботі типу метеостанції із веб-інтерфейсом. Мікроконтролер вартує мінімальних грошей, що ідеально підходить для бюджетної розробки метеостанції. Але незважаючи на невелику ціну, він не поступається у свої якості чи функціональності. Що дозволяє створювати бюджетну, високоефективну систему без значного фінансового вкладу.

Отже, вибір мікроконтролера *ESP8266* для розробки метеостанції з веб-інтерфейсом є обумовленим завдяки його вбудованому *Wi-Fi*, енергоефективності, компактним розмірам, достатній обчислювальній потужності, гнучкості в підключенні датчиків, низькій вартості, великій спільноті підтримки, простоті програмування та можливості віддаленого оновлення. Ці характеристики забезпечують створення надійної, функціональної та економічно вигідної системи, яка відповідає сучасним вимогам до моніторингу погоди та віддаленого доступу до даних через веб-інтерфейс. *ESP8266* поєднує в собі все необхідне для реалізації проекту

метеостанції, роблячи його оптимальним рішенням для розробки, яка прагне досягти високої якості при розумних витратах.

2.2 Вибір сенсорів для вимірювання параметрів

Датчик *DHT22* є популярним цифровим датчиком, який широко використовується для вимірювання як температури, так і вологості повітря одночасно, що робить його надзвичайно універсальним і привабливим для використання у різних проєктах, зокрема у складі метеостанцій, де важливо мати надійний та простий у використанні сенсор для вимірювання основних параметрів навколишнього середовища. Завдяки своїй конструкції, яка поєднує в собі два різні сенсори — для температури та для вологості — він забезпечує зручність і компактність, оскільки користувачеві не потрібно окремо підключати два різні пристрої. *DHT22* працює за цифровим принципом, що означає, що його вихідний сигнал вже є оцифрованим і готовим для обробки мікроконтролером *ESP8266* або іншим аналогічним пристроєм без необхідності використання додаткових аналогово-цифрових перетворювачів або інших компонентів. Це значно спрощує інтеграцію та розробку проєктів, оскільки цифровий інтерфейс полегшує обмін даними між сенсором та контролером. Принцип роботи датчика *DHT22*, як уже зазначалося раніше, базується на двох основних технологіях вимірювання: для вимірювання вологості він використовує невеликий сенсор, вологість визначається за допомогою конденсату, що накопичується всередині спеціальної ємності у датчику, завдяки чому датчик здатний вловлювати зміни вологості повітря з відносно високою точністю. Щодо температури, то її вимірювання здійснюється за допомогою терморезистора, опір якого змінюється в залежності від температури навколишнього середовища, дозволяючи датчику досить точно визначати температуру та передавати цю інформацію у цифровому форматі для подальшої обробки. З технічних характеристик цього датчика варто виділити його широкий діапазон

вимірювання температури, який становить від -40°C до 80°C із точністю приблизно $\pm 0.5^{\circ}\text{C}$, що дозволяє використовувати його у найрізноманітніших кліматичних умовах, починаючи від суворих зимових морозів і завершуючи спекотним літом, яке характерне для більшості регіонів України.

Завдяки такому широкому температурному діапазону датчик ідеально підходить для метеостанцій, оскільки дозволяє відстежувати погодні умови у будь-яку пору року. Що стосується вологості, то *DHT22* здатен вимірювати її у діапазоні від 0% до 100% із точністю, що варіюється від $\pm 2\%$ до $\pm 5\%$, що робить його придатним для використання у різних умовах, де рівень вологості може змінюватися від дуже сухого повітря до умов із високою вологістю, наприклад, під час дощу або в умовах підвищеної вологості у літній період.

Такий діапазон вимірювання вологості дозволяє мати уявлення про стан навколишнього середовища, що є надзвичайно важливим для систем моніторингу погоди та забезпечення достовірних даних для користувачів метеостанцій. Частота оновлення даних у *DHT22* становить приблизно 0.5 Гц, що означає, що датчик надає нові дані приблизно кожні 2 секунди, чого цілком достатньо для моніторингу погодних умов, оскільки вони, як правило, не змінюються миттєво і не потребують надто високої частоти оновлення, як у випадку деяких інших застосунків, наприклад, у промислових або критичних системах контролю.

Для метеостанцій цього цілком вистачає, оскільки погодні умови змінюються повільно, а надмірно часті вимірювання лише перевантажують систему зайвими даними без істотного приросту корисної інформації. Що стосується живлення, то *DHT22* можна живити як від 3.3 В, так і від 5 В, що робить його універсальним і сумісним із більшістю сучасних платформ, серед яких можна назвати *ESP8266*, *Arduino*, *ESP32* та інші мікроконтролери, які найчастіше використовуються для створення любительських і професійних метеостанцій. Така гнучкість у виборі напруги живлення дозволяє інтегрувати цей сенсор у різноманітні проекти без необхідності використання додаткових перетворювачів або стабілізаторів напруги. Основною перевагою *DHT22*

можна вважати його доступність і простоту у використанні, адже завдяки цифровому виходу його підключення до мікроконтролера здійснюється через один вивід даних, що значно спрощує процес розробки проєктів і усуває потребу у складних схемах підключення або використанні додаткових компонентів для обробки аналогового сигналу.

Крім того, датчик має відносно високу стабільність роботи, що означає, що він здатен працювати безперервно протягом тривалого часу, забезпечуючи постійний моніторинг погодних умов, що є важливою вимогою для метеостанцій, які зазвичай функціонують цілодобово без перерв.

Це особливо важливо для систем, де потрібне надійне та безперервне вимірювання параметрів навколишнього середовища. Ще однією важливою перевагою *DHT22* є його низька вартість, завдяки чому цей сенсор стає привабливим для використання у багатьох аматорських та професійних проєктах, особливо коли бюджет обмежений, а необхідно отримати достатньо точні дані про температуру та вологість.

Саме через це *DHT22* є ідеальним вибором для цієї кваліфікаційної роботи, оскільки він поєднує в собі всі необхідні характеристики: доступність, простоту підключення, надійність роботи, достатню точність вимірювань та універсальність застосування. Однак, як і будь-який інший сенсор, *DHT22* має свої недоліки, про які також необхідно згадати для повноти опису його характеристик.

Зокрема, точність вимірювання вологості може дещо знижуватися при екстремальних значеннях, що стає особливо помітно у дуже спекотних регіонах або під час тривалих періодів високої температури. Це може бути актуально для південних та центральних регіонів України, таких як Миколаївська, Дніпропетровська, Херсонська та Запорізька області, де влітку температура часто досягає високих позначок, що може призвести до незначних похибок у вимірюванні вологості, хоча для більшості застосувань ця похибка не є критичною. Крім того, датчик *DHT22* досить чутливий до пилу та інших забруднень, які можуть накопичуватися на його поверхні або

всередині корпусу, що потенційно може впливати на його точність та стабільність роботи. Це також варто враховувати при виборі місця встановлення датчика, адже його розміщення у місці з високим рівнем запиленості може призвести до необхідності частого обслуговування або навіть до виходу сенсора з ладу. Для запобігання подібним проблемам доцільно передбачити відповідні заходи захисту, наприклад, встановлення захисної сітки або регулярне очищення. Попри всі ці особливості, датчик *DHT22* залишається одним із найбільш популярних та затребуваних сенсорів для вимірювання температури та вологості, завдяки своїй універсальності та простоті інтеграції у різноманітні проекти, особливо у складі систем метеомоніторингу, де він демонструє надійну та стабільну роботу протягом тривалого часу. Саме завдяки цим характеристикам він був обраний для цієї кваліфікаційної роботи, у якій розробляється метеостанція на основі мікроконтролера *ESP8266*.

На рисунку 2.2 буде представлено зовнішній вигляд датчика *DHT22*, що дозволить краще зрозуміти його конструкцію та принцип роботи у складі метеостанції



Рисунок 2.2 – Датчик температури *DHT22*

Але все ж таки, у даній кваліфікаційній роботі датчик *DHT22*, буде використовуватися для вимірювання температури.

Датчик тиску *BME280* – це теж багатофункціональний датчик розроблений німецькою компанією *Bosch*. Здатен вимірювати температуру, вологість та атмосферний тиск. Вважаю ідеальним рішенням інтегрувати цей датчик у метеостанцію, оскільки можна отримати повний набір

метеорологічних показників. *BME280* базується на технології *MEMS* (мікроелектромеханічні системи), що забезпечує високу точність і надійність. Принцип роботи *BME280* складніший, ніж у *DHT22*, через його багатофункціональність. Температура вимірюється за допомогою терморезистора, який реагує на зміни тепла в навколишньому середовищі. Вологість фіксується ємнісним сенсором, який змінює свої властивості залежно від кількості вологи в повітрі.

Атмосферний тиск вимірюється п'єзорезистивним сенсором: тиск повітря деформує спеціальну мембрану, що впливає на опір внутрішніх компонентів.

Усі ці дані обробляються вбудованим процесором і передаються через інтерфейси *I2C* або *SPI*, що робить датчик гнучким для підключення. На рисунку 2.3 зображено датчик *BME280*.



Рисунок 2.3 – Датчик *BME280*

Перейдемо до технічних характеристик цього датчика. Хоча для вимірювання температури він не буде використовуватися, але все ж таки це може знадобитися. Тож, температура вимірюється в діапазоні від -40°C до 85°C із точністю $\pm 1^{\circ}\text{C}$, що підходить для більшості погодних умов. Вологість фіксується з точністю $\pm 3\%$ у діапазоні від 0% до 100%, що перевершує показники багатьох аналогів. Атмосферний тиск вимірюється в діапазоні від 300 до 1100 гПа з точністю ± 1 гПа. Така точність дозволяє використовувати *BME280* для прогнозування погоди та навіть визначення висоти над рівнем моря. Енергоспоживання датчика становить лише 3.6 мкА при частоті 1 Гц, що робить його енергоефективним. З переваг його найперших – це

універсальність, компактність завдяки якій дуже легко його вмонтувати у портативну систему. Ну і низьке енергоспоживання, тому що може працювати від звичайної батареї.

Датчик тиску *BMP280*. Це спеціалізований датчик атмосферного тиску, також розроблений компанією *Bosch*. На відміну від *BME280*, він не вимірює вологість, але зберігає високу точність у фіксації тиску та температури. Але буде використовуватись для вимірювання тиску. Базується на п'єзорезистивній технології, що забезпечує стабільні та точні вимірювання. Принцип роботи *BMP280* полягає в тому, що тиск повітря впливає на деформацію внутрішньої мембрани, змінюючи опір п'єзорезистивних елементів.

Температура вимірюється для компенсації впливу на показники тиску, що підвищує точність. Дані передаються через інтерфейси *I2C* або *SPI*, що дозволяє легко підключати датчик до мікроконтролерів. Компактність і простота роблять *BMP280* популярним вибором для багатьох застосувань. Технічні параметри *BMP280* охоплюють проміжок тиску від 300 до 1100 гПа з точністю до ± 1 гПа. Це дає змогу реєструвати навіть незначні коливання, які можуть вказувати на наближення бурі або ясну погоду.

Температура вимірюється у діапазоні від -40°C до 85°C з точністю $\pm 1^{\circ}\text{C}$. Енергетичне споживання складає лише 2.7 мкА при частоті 1 Гц, а габарити сенсора – 2.0 x 2.5 мм. Цей комплекс характеристик робить його відповідним для автономних та портативних систем. Переваги *BMP280* полягають у високій точності вимірювань та його економічності. Вбудована температурна компенсація гарантує стабільні значення тиску за різних умов навколишнього середовища. Датчик також характеризується високою швидкістю вимірювання (до 157 Гц), хоча для метеорологічних станцій зазвичай вистачає меншої частоти.

У порівнянні з *BME280*, *BMP280* коштує дешевше, що робить його привабливим для проектів, де вимірювання вологості не є визначальним фактором. Датчик буде зображений на рисунку 2.4.



Рисунок 2.4. – Датчик тиску *BMP280*

Наступним датчиком буде *LDR* – простий датчик, призначений для вимірювання рівня освітленості. В даній кваліфікаційній роботі, одне із найдешевших рішень для моніторингу світла на метеостанції. Його принцип роботи, побудований на фотопровідності, тобто опір резистора буде зменшуватися при збільшенні інтенсивності світла. А у темряві, опір може сягати мегаОмів. Переваги *LDR* криються в його простоті та невисокій ціні. Це один з найдешевших датчиків для вимірювання рівня освітлення, що робить його надзвичайно популярним для проектів з обмеженим бюджетом. *LDR* легко інтегрується в системи і не вимагає складного калібрування. Його компактні розміри дозволяють використовувати датчик у невеликих метеостанціях, не збільшуючи загальні габарити. Недоліки *LDR* включають обмежену точність. Зміна опору має нелінійний характер, що ускладнює отримання достовірних даних без подальшої обробки.

Датчик чутливий до всього видимого спектру світла, що може стати мінусом, якщо потрібно вимірювати конкретні довжини хвиль. Час відгуку *LDR* (десятки мілісекунд) є повільнішим, ніж у цифрових сенсорів, але для метеостанцій це рідко буває критичним.

2.3 Огляд технологій для створення веб-інтерфейсу

Розробка веб-інтерфейсу є ключовим етапом у створенні сучасних автоматизованих систем, які забезпечують зручний доступ до даних і

дозволяють користувачам ефективно взаємодіяти з пристроями. У контексті кваліфікаційної роботи, що передбачає інтеграцію з мікроконтролером *ESP8266* та датчиками, такими як *DHT22* для вимірювання температури і вологості, *BME280* для моніторингу вологості, температури та тиску, *BMP280* для фіксації атмосферного тиску та *LDR* для визначення рівня освітленості, веб-інтерфейс відіграє роль центрального інструменту, який об'єднує апаратне забезпечення з потребами користувача. Веб-інтерфейс має бути не лише функціональним, але й інтуїтивно зрозумілим, адаптивним і здатним працювати в реальному часі, щоб відображати актуальні дані з датчиків.

Цей огляд розглядає основні технології, які можуть бути використані для створення такого інтерфейсу, аналізує їхні можливості, особливості та специфіку застосування з урахуванням вимог до роботи з мікроконтролером і датчиками. *HTML (HyperText Markup Language)* є фундаментом будь-якого веб-інтерфейсу, оскільки визначає його структуру та зміст. Ця технологія дозволяє створювати елементи сторінки, такі як заголовки, таблиці, блоки для відображення числових значень чи графіків, що є необхідним для представлення даних із датчиків. Наприклад, показники температури з *DHT22* чи тиску з *BMP280* можна відобразити в окремих секціях сторінки, структурованих за допомогою тегів *div* чи *table*. Використання *HTML5* розширює можливості завдяки підтримці семантичних тегів, таких як *header*, *footer* чи *section*, що покращує читабельність коду та полегшує індексацію сторінки пошуковими системами. Крім того, *HTML5* підтримує вбудовані мультимедійні елементи, наприклад, *canvas* для створення динамічних графіків, які можуть візуалізувати зміни вологості чи освітленості з плином часу. Простота *HTML* дозволяє швидко створювати базовий інтерфейс, який може бути розширений за допомогою інших технологій для забезпечення інтерактивності та стилізації. *CSS (Cascading Style Sheets)* відповідає за зовнішній вигляд веб-інтерфейсу, дозволяючи задавати кольори, шрифти, розміри елементів і їх розташування. У кваліфікаційній роботі, де інтерфейс має бути зручним для перегляду даних із датчиків, *CSS* допомагає створювати

чіткий і привабливий дизайн. Наприклад, можна виділити показники температури яскравим кольором, а значення тиску відобразити в іншому стилі, щоб користувач одразу звертав увагу на ключові параметри. Використання фреймворків, таких як *Bootstrap* або *Tailwind CSS*, значно спрощує створення адаптивного дизайну, який коректно відображається на екранах різного розміру — від настільних комп'ютерів до мобільних телефонів. Адаптивність є особливо важливою, адже користувачі можуть отримувати доступ до метеостанції через смартфони, перевіряючи дані про погоду в реальному часі. *CSS* також дозволяє застосовувати анімації, наприклад, плавне оновлення значень чи підсвічування при зміні освітленості, що робить інтерфейс більш інтерактивним.

Завдяки модульності *CSS* стилі можна легко змінювати, не впливаючи на структуру сторінки, що полегшує підтримку та вдосконалення інтерфейсу в процесі розробки. *JavaScript* є основною технологією для створення інтерактивних елементів веб-інтерфейсу. Він дозволяє обробляти дані, отримані від мікроконтролера *ESP8266*, і динамічно оновлювати їх на сторінці без необхідності її перезавантаження. Наприклад, *JavaScript* може періодично надсилати запити до *ESP8266*, щоб отримати актуальні значення температури з *DHT22* або тиску з *BME280*, і відобразити їх у реальному часі.

Технологія *AJAX* (*Asynchronous JavaScript and XML*) забезпечує асинхронну взаємодію з сервером, що дозволяє оновлювати лише частину сторінки, наприклад, графік із даними вологості. Бібліотеки, такі як *Chart.js*, полегшують створення візуально привабливих графіків і діаграм, які можуть показувати динаміку зміни параметрів за годину, день чи тиждень.

Інша бібліотека, *D3.js*, пропонує ще більшу гнучкість для створення складних візуалізацій, таких як теплові карти чи 3D-графіки, хоча її використання потребує глибшого розуміння програмування. *JavaScript* також дозволяє реалізувати інтерактивні елементи, наприклад, кнопки для увімкнення чи вимкнення датчиків або фільтри для вибору періоду відображення даних. Завдяки універсальності *JavaScript* інтерфейс може бути

адаптований до специфічних потреб користувачів, забезпечуючи гнучкість і функціональність.

Для створення складніших і масштабованих інтерфейсів доцільно використовувати фронтенд-фреймворки, такі як *React*, *Angular* або *Vue.js*. *React*, розроблений *Facebook*, дозволяє створювати компонентно-орієнтовані інтерфейси, де кожен елемент, наприклад, блок із даними про температуру чи вологість, є окремим компонентом. Це полегшує підтримку коду та додавання нових функцій, таких як відображення історичних даних із *BME280*. *Vue.js* є легшим у освоєнні і підходить для швидкої розробки простих, але ефективних інтерфейсів. *Angular*, своєю чергою, пропонує комплексне рішення з потужними інструментами для створення великих систем, хоча його складність може бути надмірною для кваліфікаційної роботи. Усі ці фреймворки підтримують інтеграцію з *API*, що дозволяє отримувати дані від *ESP8266* і відображати їх у реальному часі. Вибір фреймворку залежить від рівня складності інтерфейсу: для базового рішення, яке відображає лише поточні значення датчиків, достатньо чистого *JavaScript* із бібліотеками, тоді як для створення системи з розширеними функціями, наприклад, прогнозуванням погоди, *React* або *Vue.js* будуть більш доречними.

Мікроконтролер *ESP8266* із вбудованим модулем *Wi-Fi* може функціонувати як веб-сервер, що робить його самодостатнім рішенням для створення веб-інтерфейсу. Використання бібліотеки *ESP8266WebServer* дозволяє налаштувати сервер, який обробляє *HTTP*-запити від браузера і повертає дані з датчиків у форматі *JSON* або *HTML*. Наприклад, *ESP8266* може надсилати значення температури з *DHT22* чи тиску з *BMP280* у відповідь на запит користувача, а веб-інтерфейс відобразить їх у зрозумілій формі. Таке рішення є економічним, оскільки не потребує додаткового обладнання, і простим у реалізації, адже бібліотека містить готові функції для роботи з *HTTP*. Однак ресурси *ESP8266* обмежені: обсяг пам'яті та обчислювальна потужність дозволяють створювати лише базові інтерфейси з мінімальною кількістю сторінок і функцій. Для відображення складних графіків чи

збереження великих обсягів даних із датчиків може знадобитися зовнішній сервер. Для створення більш потужного та масштабованого веб-інтерфейсу доцільно використовувати зовнішній сервер, який обробляє дані від *ESP8266* і передає їх у браузер. Технології, такі як *Node.js*, є популярними завдяки своїй асинхронній архітектурі, яка ідеально підходить для обробки даних у реальному часі. Наприклад, сервер на *Node.js* може отримувати показники вологості з *BME280* через *HTTP*-запити і зберігати їх у базі даних для подальшого аналізу. Фреймворки *Express* для *Node.js* спрощують створення *REST API*, через які веб-інтерфейс може отримувати структуровані дані. Альтернативою є *Python* із фреймворками *Flask* або *Django*. *Flask* є легким і підходить для швидкої розробки *API*, тоді як *Django* пропонує більше вбудованих інструментів, таких як адмін-панель для управління даними. Зовнішній сервер дозволяє реалізувати складні функції, наприклад, прогнозування погоди на основі історичних даних із *BMP280*, але потребує додаткових ресурсів, таких як хостинг або локальний комп'ютер.

WebSocket є потужною технологією для забезпечення оновлення даних у реальному часі, що є критично важливим для відображення показників із датчиків, таких як *LDR* чи *DHT22*. На відміну від *HTTP*, де клієнт надсилає запит, а сервер відповідає, *WebSocket* встановлює постійне двостороннє з'єднання. Це дозволяє *ESP8266* автоматично надсилати нові дані, наприклад, при зміні рівня освітленості, а веб-інтерфейс миттєво їх відображає. Бібліотека *WebSocketServer* для *ESP8266* полегшує реалізацію такого підходу, хоча обмежені ресурси мікроконтролера можуть ускладнити обробку великої кількості підключень. У разі використання зовнішнього сервера, наприклад, на *Node.js*, *WebSocket* працює ефективніше, підтримуючи десятки чи сотні клієнтів одночасно. *REST API* (*Representational State Transfer*) є стандартним підходом для організації взаємодії між веб-інтерфейсом і сервером. У кваліфікаційній роботі це може бути реалізовано через створення *endpoints*, таких як */temperature* для даних із *DHT22* або */pressure* для *BMP280*. Веб-інтерфейс періодично надсилає *GET*-запити до цих адрес, отримуючи

актуальні значення у форматі *JSON*. *ESP8266* може виступати сервером для *REST API*, використовуючи бібліотеку *ESP8266WebServer*, або передавати дані на зовнішній сервер, який обробляє запити. *REST API* є простим у реалізації та універсальним, оскільки підтримується більшістю сучасних технологій. Однак для оновлення даних у реальному часі потрібні часті запити, що може створювати додаткове навантаження на мікроконтролер. *MQTT* (*Message Queuing Telemetry Transport*) є легким протоколом для обміну даними, який ідеально підходить для систем із обмеженими ресурсами, таких як *ESP8266*. У цьому підході мікроконтролер виступає клієнтом, який публікує дані з датчиків на брокер *MQTT*, наприклад, *Mosquitto*. Веб-інтерфейс, своєю чергою, підписується на відповідні теми, отримуючи оновлення в реальному часі. Наприклад, дані про вологість із *BME280* можуть надсилатися на тему */humidity*, а інтерфейс автоматично відобразить їх. *MQTT* економить ресурси порівняно з *HTTP* забезпечує швидку передачу даних, що робить його ефективним для метеостанцій.

РОЗДІЛ 3

РОЗРОБКА МЕТЕОСТАНЦІЇ

3.1 Архітектура системи та схема підключення компонентів

Метеостанція, яка створюється в межах даної кваліфікаційної роботи, є прикладом сучасного вбудованого пристрою, що інтегрує сенсорні технології, мікроконтролерну логіку та бездротову передачу даних. Основною функцією системи є цілодобовий моніторинг метеорологічних параметрів навколишнього середовища, таких як температура повітря, відносна вологість, атмосферний тиск та рівень освітленості. Для досягнення цієї мети розроблена архітектура, заснована на концепції «інтелектуального вузла», де обробка даних та їх передача здійснюються централізовано за допомогою одного компактного, проте достатньо потужного обчислювального блоку — мікроконтролера *NodeMCU* з чипом *ESP8266*.

Архітектура метеостанції передбачає підключення декількох типів сенсорів, які взаємодіють із *NodeMCU*. Загалом, усі сенсори можна умовно поділити на цифрові та аналогові залежно від способу передавання даних. Цифрові сенсори надсилають інформацію у вигляді послідовності сигналів через спеціалізовані інтерфейси, тоді як аналогові формують напругу, пропорційну вимірюваному параметру. Центральне місце серед сенсорів посідає високоточний модуль *BME280* виробництва компанії *Bosch Sensortec*.

Цей пристрій забезпечує одночасне вимірювання трьох ключових параметрів — температури повітря, відносної вологості та атмосферного тиску. Завдяки такій багатофункціональності, *BME280* дозволяє скоротити кількість окремих елементів у схемі та оптимізувати конструкцію пристрою. Для обміну даними з мікроконтролером застосовується інтерфейс *I2C* — двопровідна шина, яка дає змогу підключати декілька пристроїв до однієї пари ліній. У даному випадку лінія синхронізації (*SCL*) підключена до виводу *D1* (*GPIO 5*), а лінія передачі даних (*SDA*) — до виводу *D2* (*GPIO 4*) плати

NodeMCU. Живлення сенсора здійснюється від виводу 3.3V, що відповідає стандартному рівню напруги для *ESP8266*, оскільки перевищення цього значення може призвести до виходу мікросхеми з ладу. Контакт *GND* об'єднано з відповідним виводом на платі.

Адреса *BME280* за замовчуванням становить 0x77, що використовується для його ідентифікації на шині *I2C*. Паралельно в системі застосовується сенсор *BMP280*, який є спрощеною версією *BME280*, не має функції вимірювання вологості, проте забезпечує високоточне вимірювання температури та тиску. Це дає змогу реалізувати додатковий рівень перевірки даних та підвищити точність вимірювань. *BMP280* підключається до тих самих ліній *I2C* (*SCL* — *D1*, *SDA* — *D2*), оскільки його адреса 0x76 відрізняється від адреси *BME280*.

Таким чином, обидва сенсори можуть працювати одночасно на одній шині без виникнення конфліктів. Для переведення *BMP280* у режим *I2C* необхідно підключити вивід *CSB* до 3.3V, а *SDO* — до *GND*. Така схема є стабільною та підтвердженою практикою багатьох розробників. Ще одним важливим елементом є сенсор *DHT22* — цифровий датчик, що вимірює температуру та вологість із високою точністю. Для передачі інформації він використовує лише один вихідний сигнал (*DATA*), який надсилає закодовану інформацію до мікроконтролера. У даному випадку контакт *DATA* підключено до виводу *D5* (*GPIO 14*). Живлення здійснюється через вивід 3.3V, а земля під'єднана до *GND*. Особливістю підключення *DHT22* є необхідність використання підтягувального резистора номіналом 4.7 кОм між лініями *VCC* та *DATA*. Це забезпечує стабільний логічний рівень і мінімізує ризик спотворення сигналу.

Такий сенсор може працювати як резервний або дублюючий елемент системи, а також використовуватися для перевірки коректності даних від *BME280* у випадках різкої зміни умов навколишнього середовища. Для вимірювання рівня освітленості середовища використовується аналоговий фоторезистор *LDR*, який змінює свій опір залежно від інтенсивності світла.

Його підключення реалізовано через подільник напруги: *LDR* з'єднується послідовно з резистором номіналом 10 кОм, що підключений до землі (*GND*). Інший кінець *LDR* підключено до живлення 3.3V, а точка поділу між ними — до аналогового входу *A0* на *NodeMCU*. Таким чином, на *A0* подається напруга, що змінюється пропорційно до освітленості середовища. Така схема дозволяє системі в реальному часі зчитувати рівень освітленості та, за потреби, реагувати на нього — наприклад, зберігати значення у лог або автоматично змінювати режим роботи пристрою залежно від часу доби чи пори року. На рисунку 3.1 зображено схему будовання даної метеостанції.

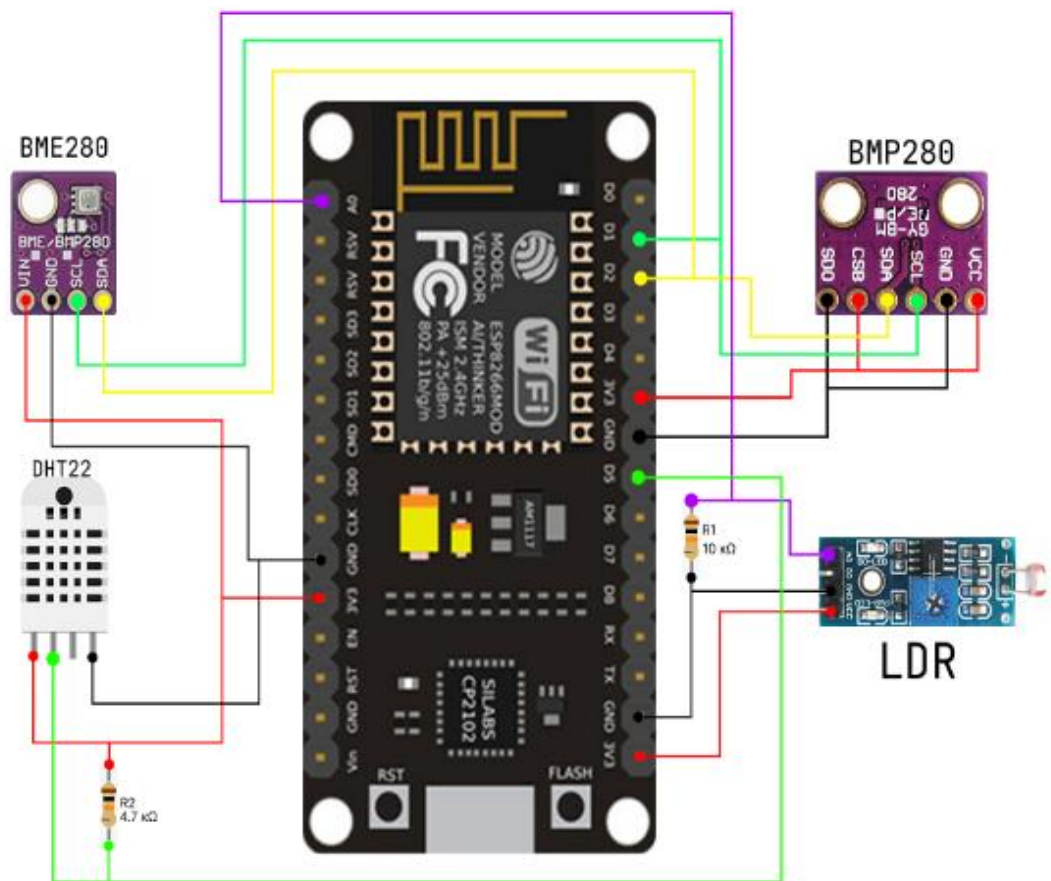


Рисунок 3.1 – Схема будовання метеостанції

3.2 Програмування мікроконтролера *ESP*

У рамках мого проєкту я реалізував метеостанцію з веб-інтерфейсом на базі мікроконтролера *ESP8266*, яка призначена для збору, обробки та

візуалізації кліматичних даних у режимі реального часу. Програмування мікроконтролера стало ключовим етапом розробки, адже саме воно визначає логіку роботи пристрою, взаємодію з сенсорами та організацію доступу до даних через мережу. У моєму випадку мова йде про використання мікроконтролера *NodeMCU* (на базі *ESP8266*), який має вбудований *Wi-Fi* модуль, що дає змогу організувати бездротову передачу даних без використання додаткових мережевих модулів. Даний контролер підтримує популярне середовище розробки *Arduino IDE*, що дозволяє писати код мовою *C++* із застосуванням відповідних бібліотек для керування сенсорами та організації веб-сервера. Програмування мікроконтролера розпочиналося з ініціалізації підключення сенсорів та підключення необхідних бібліотек. Зокрема, у даній кваліфікаційній роботі використовувалися такі бібліотеки, як *ESP8266WiFi* для роботи з мережею, *ESP8266WebServer* для створення та обробки запитів *HTTP*, а також бібліотеки для роботи з сенсорами — *Adafruit BME280* та *DHT*. Кожна з цих бібліотек виконує окрему функцію: забезпечує підключення до *Wi-Fi*, організовує обробку запитів від користувача, а також реалізує зчитування температури, вологості, атмосферного тиску, висоти над рівнем моря та рівня освітленості.

Для взаємодії з шиною *I2C* було задіяно бібліотеку *Wire*. Завдяки цій бібліотеці мікроконтролер може здійснювати виявлення підключених до шини пристроїв та отримувати від них необхідні дані. У програмному коді передбачена функція сканування *I2C*-пристроїв, яка дозволяє на етапі запуску мікроконтролера визначити, чи доступні підключені сенсори, зокрема *BME280*. Це є важливим аспектом, оскільки можливе використання сенсора з різною адресою (*0x76* або *0x77*), і у випадку його відсутності програма виводить відповідне повідомлення.

Окрім сенсора *BME280*, у даній кваліфікаційній роботі також використовується сенсор *DHT22*, який передає дані про температуру та вологість повітря, а також фоторезистор (*LDR*), підключений до аналогового входу мікроконтролера для вимірювання рівня освітленості в навколишньому

середовищі. Ключовою особливістю реалізації даної метеостанції є наявність вбудованого веб-сервера, який запускається при старті мікроконтролера. Його налаштування виконано у режимі точки доступу (*Access Point*), що дає змогу користувачам підключатися до мережі *ESP8266-AP* безпосередньо, без необхідності використання зовнішнього роутера.

Такий підхід значно спрощує структуру мережі та мінімізує кількість додаткового обладнання, необхідного для роботи метеостанції. Після підключення до мережі користувач має змогу відкрити веб-інтерфейс у браузері, який розміщено безпосередньо на мікроконтролері, та переглядати всі зібрані дані в режимі реального часу. У програмному забезпеченні веб-сервер обробляє два основні маршрути: кореневу сторінку ("/"), яка відповідає за формування та повернення *HTML*-сторінки з вбудованими стилями та скриптами, а також маршрут *"/data"*, який відповідає за надання даних у форматі *JSON* у відповідь на запити клієнта.

Така структура дозволяє розділити логіку між відображенням інтерфейсу користувача та отриманням актуальних даних від сенсорів. Завдяки використанню *JavaScript* на стороні клієнта реалізовано автоматичне оновлення даних кожні кілька секунд без необхідності перезавантаження сторінки, що підвищує інтерактивність та зручність користування системою.

У коді мікроконтролера кожні три секунди зчитуються значення температури, вологості, атмосферного тиску, висоти над рівнем моря та освітленості. Отримані дані формуються у вигляді *JSON*-об'єкта та передаються у відповідь на *HTTP*-запит користувача. Такий підхід забезпечує динамічну роботу системи, дозволяючи відстежувати зміну кліматичних параметрів у режимі реального часу та своєчасно реагувати на зміни навколишнього середовища. На рисунку 3.2 буде зображено фрагмент коду для обробки даних сенсорів та формування відповіді у форматі *JSON*.

```

server.on("/data", []() {
    float temp = NAN, hum = NAN, pres = NAN, alt = NAN;
    int ldr = analogRead(LDRPIN);

    if (bme.begin(0x76) || bme.begin(0x77)) {
        temp = bme.readTemperature();
        hum = bme.readHumidity();
        pres = bme.readPressure() / 100.0F;
        alt = bme.readAltitude(1013.25);
    }

    float dhtTemp = dht.readTemperature();
    float dhtHum = dht.readHumidity();

    if (!isnan(dhtTemp)) temp = dhtTemp;
    if (!isnan(dhtHum)) hum = dhtHum;

    staticJsonDocument<200> doc;
    doc["temp"] = temp;
    doc["hum"] = hum;
    doc["pres"] = pres;
    doc["alt"] = alt;
    doc["ldr"] = ldr;

    String output;
    serializeJson(doc, output);
    server.send(200, "application/json", output);
});

```

Рисунок 3.2 – Фрагмент коду для обробки даних сенсорів

Дані з сенсорів обробляються окремо: при наявності модуля *BME280* зчитуються його показники, у протилежному випадку ці параметри відображаються як «недоступні» (*NaN*). Це дозволяє зберігати стабільність

роботи пристрою навіть у випадку несправностей окремих компонентів. Для сенсора *DHT22* окремо викликаються функції читання температури та вологості, а аналоговий фоторезистор дає значення освітленості у вигляді цілого числа. *HTML*-сторінка, що генерується сервером, включає сучасний адаптивний інтерфейс із розміщенням показників у вигляді окремих блоків-карток. У дизайні використано градієнтне тло, прозорі панелі з тінями та зручну сітку для розміщення елементів. Завдяки *JavaScript*-скриптам, дані на сторінці оновлюються динамічно, без потреби у ручному оновленні, що значно підвищує зручність користувача.

Аналіз предметної області показав, що доцільніше використати максимально простий, але ефективний дизайн, який дозволяє зосередитися на функціональності та зменшити розмір коду, що зберігається в пам'яті мікроконтролера. Однією з особливостей мікроконтролера *ESP8266* є обмежені ресурси — як оперативної пам'яті, так і місця для зберігання програм. Тому важливо оптимізувати код, уникати зайвих обчислень і виводів, а також правильно управляти структурою даних.

В проєкті використовував *String*-об'єкти обережно, намагаючись мінімізувати створення тимчасових копій рядків, щоб уникнути фрагментації пам'яті. Крім того, при форматуванні чисел я задавав кількість знаків після коми, щоб уникати зайвого навантаження при передаванні великих обсягів даних. Також важливим моментом є наявність логів через порт *UART*, які дають змогу бачити поточний стан пристрою, *IP*-адресу точки доступу, виявлені пристрої *I2C* та значення сенсорів у реальному часі. Ця інформація незамінна під час налагодження системи, перевірки коректності роботи обладнання та аналізу можливих помилок. Усі повідомлення виводяться з поясненнями, що спрощує аналіз навіть без глибокого занурення в код. У процесі розробки я зіткнувся з низкою викликів, зокрема з коректною ініціалізацією сенсора *BME280*. Через те, що він може бути налаштований на різні *I2C*-адреси, довелося реалізувати перевірку одразу для обох можливих адрес. Також у роботі з *DHT22* необхідно враховувати затримки при

зчитуванні даних, оскільки цей сенсор досить чутливий до часу між запитами, і надто часті виклики можуть призводити до помилок або отримання некоректних значень. Окрему увагу я приділив стабільності роботи веб-сервера. В умовах обмежених ресурсів *ESP8266* обробка *HTTP*-запитів може призводити до перевантаження, тому я налаштував сервер так, щоб одночасно обслуговувати лише базову кількість клієнтів. Також передбачено обробку випадків, коли користувач переходить на неіснуючі сторінки — у цьому разі сервер повертає повідомлення про помилку 404 із поясненням, що сторінку не знайдено.

Узагальнюючи, можна сказати, що програмування мікроконтролера — це не лише написання коду, а повноцінна інженерна діяльність, яка охоплює налаштування обладнання, тестування, оптимізацію ресурсів і створення взаємодії між апаратною частиною та кінцевим користувачем. Завдяки мікроконтролеру *ESP8266* мені вдалося реалізувати повноцінну автономну метеостанцію, яка не потребує зовнішніх серверів і забезпечує зручний інтерфейс доступу до кліматичних даних у будь-який момент часу. Результатом мого програмування стала стабільна система, яка демонструє ефективність використання мікроконтролерів у проєктах Інтернету речей. Такий підхід може бути масштабований або адаптований до інших сценаріїв — наприклад, для моніторингу якості повітря, систем автоматизації вдома, або навчальних лабораторних стендів.

Метеостанція є прикладом того, як за допомогою доступних мікроконтролерів і відкритого програмного забезпечення можна створити складний, але водночас зрозумілий і функціональний пристрій. Саме такі рішення мають велике значення для освіти, наукових досліджень і побудови сучасних систем моніторингу навколишнього середовища.

3.3 Збір та обробка метеоданих

У даному розділі детально розглядається реалізація одного з ключових функціональних компонентів системи — процесу збору, обробки, збереження та візуалізації метеорологічних даних, що формуються на основі показників, які надходять із набору сенсорів, підключених до мікроконтролера *ESP8266*. Мета цієї частини проєкту полягає у створенні автономного, надійного та максимально точного вузла вимірювання параметрів навколишнього середовища, що здатен у режимі реального часу надавати кінцевому користувачеві відомості про погодні умови, що переважають у конкретному місці встановлення пристрою.

У даній кваліфікаційній роботі ця система може стати базовим елементом для побудови більш складної розподіленої мережі метеостанцій, які можуть синхронізовано обробляти, накопичувати та передавати дані до центрального вузла, створюючи тим самим умови для аналізу погодних змін у різних регіонах із застосуванням елементів штучного інтелекту або машинного навчання. У рамках цього етапу реалізації було використано кілька сенсорних модулів, кожен з яких виконує свою специфічну функцію. Основним джерелом даних про мікроклімат виступає датчик *BME280*, який здатен одночасно фіксувати температуру повітря, його відносну вологість, а також барометричний тиск. На основі останнього, завдяки наявності відповідного математичного модуля, також обчислюється висота над рівнем моря, що дозволяє з одного сенсора отримати одразу чотири важливі параметри. Висока точність вимірювання, стабільність, цифровий інтерфейс, а також компактні розміри цього модуля роблять його надзвичайно зручним для використання в мікроконтролерних проєктах, орієнтованих на довготривале автономне функціонування. З метою забезпечення надлишковості, тобто резервного дублювання показників температури та вологості, додатково було застосовано датчик *DHT22*. Хоча він має дещо нижчу точність порівняно з *BME280*, він забезпечує достатню стабільність і

дозволяє проводити порівняння результатів з метою підвищення достовірності одержуваних метеопказників. Це особливо важливо в контексті проєктів, де критичне значення має безперервність та достовірність зібраної інформації. Крім того, застосування двох незалежних сенсорів дозволяє реалізувати елементи діагностики несправностей, коли розходження між показниками перевищує допустиме значення, система може сигналізувати про можливу некоректну роботу одного з елементів.

Ще одним компонентом, що дозволяє розширити спектр фіксованих параметрів навколишнього середовища, є фоторезистивний сенсор, який дозволяє отримувати орієнтовне значення рівня освітленості. Цей параметр є надзвичайно важливим у контексті довготривалих метеоспостережень, оскільки він дозволяє зафіксувати моменти зміни світлового дня, виявити хмарність або початок опадів. Дані, отримані від цього сенсора, також можуть використовуватись у майбутньому для автоматичного регулювання енергоспоживання пристрою або для побудови прогнозів. У дальній кваліфікаційній роботі ці показники можуть бути розширені за рахунок впровадження сенсорів ультрафіолетового випромінювання, концентрації твердих частинок або газів, що дозволить трансформувати метеостанцію в повноцінний екологічний модуль моніторингу довкілля.

Програмна частина збору та обробки даних побудована за подієвою моделлю, де мікроконтролер періодично виконує зчитування даних з усіх сенсорів, перетворює їх у зручний для аналізу формат і формує на основі цього структури, які згодом передаються до клієнта через вебінтерфейс. Для цього у пам'яті мікроконтролера розгорнуто легкий *HTTP*-сервер, здатний обробляти одночасно декілька запитів.

Кожен клієнт, який звертається до пристрою за допомогою браузера, бачить у відповідь адаптивну *HTML*-сторінку, яка автоматично підлаштовується під розмір екрана пристрою, незалежно від того, чи це смартфон, ноутбук або планшет. Інтерфейс містить актуальні значення метеопказників, що автоматично оновлюються кожні кілька секунд без

потреби в ручному перезавантаженні сторінки. Водночас система реалізована у такий спосіб, що передача даних можлива як у вигляді *HTML*, так і в форматі *JSON*, що відкриває можливості для подальшої інтеграції з мобільними додатками, хмарними платформами або іншими системами збору даних. Значною перевагою створеної системи є її здатність до автономного функціонування: пристрій не потребує підключення до глобальної мережі Інтернет для передачі або відображення даних, що робить його придатним для використання в польових умовах, у віддалених районах або за відсутності стабільного доступу до мережі.

Вся обробка інформації відбувається безпосередньо на борту *ESP8266*, що мінімізує затримки та виключає необхідність зовнішніх серверів. Крім того, у процесі обробки враховуються типові похибки вимірювань, які компенсуються шляхом усереднення, накладання фільтрів або коригування, що базується на емпірично встановлених коефіцієнтах.

Усі дані, що збираються, дублюються у консоль серійного моніторингу, що дає змогу розробнику відстежувати роботу пристрою в режимі налагодження, оцінювати стабільність зчитування, виявляти потенційні збої або некоректні значення. Це також дозволяє у подальшому реалізувати збереження історії метеоспостережень на зовнішні носії, такі як карта пам'яті, або передавати їх через інтерфейси зв'язку, наприклад, *Wi-Fi* або *LoRa*, у центральну базу даних.

Отже, реалізована система збору та обробки метеоданих є не лише технічно завершеною, а й модульною та масштабованою, що дозволяє вносити зміни до конфігурації сенсорів, інтерфейсів передачі або способів обробки без необхідності повної перебудови системи.

Така гнучкість особливо цінна у проєктах навчального й дослідницького характеру, де вимагається часта адаптація під нові завдання або умови. У підсумку, створений функціональний модуль збору й обробки метеоданих повністю відповідає технічним вимогам до сучасних систем моніторингу навколишнього середовища, демонструє ефективне поєднання сенсорної

технології та мікроконтролерної обробки, а також закладає надійний фундамент для подальших науково-прикладних розробок у сфері автоматизації екологічного моніторингу, розумного довкілля та інтернету речей.

Його розгортання та тестування підтвердили стабільність, точність та функціональність рішення, що є основою для подальшої розробки масштабованих систем, які можуть бути використані як у побуті, так і в промислових або дослідницьких умовах, а згодом і в рамках ширшого застосування у дальній кваліфікаційній роботі з розробки інтегрованої мережі метеоспостереження. На рисунку 3.3 представлено частину коду, яка відповідає за дію.

```
float temp = NAN, hum = NAN, pres = NAN, alt = NAN;
int ldr = analogRead(LDRPIN);
// Спроба ініціалізувати BME280 за двома можливими адресами
if (bme.begin(0x76) || bme.begin(0x77)) {
    temp = bme.readTemperature();
    hum = bme.readHumidity();
    pres = bme.readPressure() / 100.0F;
    alt = bme.readAltitude(1013.25);
}
// Зчитування даних із DHT22
float dhtTemp = dht.readTemperature();
float dhtHum = dht.readHumidity();
// Пріоритет даних DHT над BME280
if (!isnan(dhtTemp)) temp = dhtTemp;
if (!isnan(dhtHum)) hum = dhtHum;
```

Рисунок 3.3 – Фрагмент коду відповідальний за збір даних

3.4 Реалізація веб-інтерфейсу для візуалізації показників

У даній кваліфікаційній роботі реалізація веб-інтерфейсу для візуалізації показників метеорологічного характеру на основі сенсорів, з'єднаних із мікроконтролером *ESP8266*, має ключове значення для досягнення цілей повної автономності та зручності у використанні системи моніторингу.

Основна ідея реалізації веб-інтерфейсу полягає в тому, щоб користувач міг переглядати всі актуальні дані в режимі реального часу без потреби у встановленні додаткового програмного забезпечення або складному налаштуванні клієнтської частини. Все, що потрібно користувачеві, — це доступ до *Wi-Fi*-мережі, у якій працює сам мікроконтролер, та веб-браузер, який виступає у ролі вікна доступу до зібраної сенсорами інформації. Це дозволяє мінімізувати технічні вимоги до користувача та створює інтуїтивно зрозумілий інтерфейс взаємодії з пристроєм. Невеликий фрагмент коду, який відповідає за оновлення показників зображено на рисунку 3.4.

```
<script>
  async function updateData(){
    const resp = await fetch('/data');
    const js = await resp.json();
    document.getElementById('t1').innerText =
js.T_bme.toFixed(1)+'°C';
    // ... інші значення ...
  }
  updateData();
  setInterval(updateData, 3000);
</script>
```

Рисунок 3.4 – Частина коду яка відповідає за оновлення показників

Сам *ESP8266* функціонує як веб-сервер, що реагує на вхідні *HTTP*-запити, генерує *HTML*-документ із вбудованими стилями *CSS* і сценаріями *JavaScript* та повертає його клієнту. Такий підхід дозволяє уникнути необхідності використання сторонніх веб-серверів і спрощує архітектуру

системи. Важливо, що *HTML*-сторінка, яка формується у пам'яті мікроконтролера, містить структуровані текстові блоки, у яких знаходяться місця для підставлення актуальних значень температури, вологості, атмосферного тиску, освітленості та інших параметрів. Ці значення зчитуються із сенсорів типу *BME280*, *DHT22* та фоторезистора, після чого передаються до *JavaScript*-коду у форматі *JSON*-об'єкта, який формується на окремому запиті (наприклад, за маршрутом */data*). Для ще одного прикладу на рисунку 3.5 представлено маленький фрагмент коду, який відповідає за одну візуальну картку з показником.

```
<div class="card">
  <div id="t1" class="value">--.-°C</div>
  <div class="label">Темп. №1</div>
</div>
```

Рисунок 3.5 – Фрагмент коду за одну візуальну картку з показником

Така реалізація дозволяє ефективно розділити логіку відображення даних і збір показників, що спрощує обслуговування системи. Веб-сторінка містить відповідні елементи з унікальними ідентифікаторами, які обираються скриптом на стороні клієнта. Їхній текст змінюється відповідно до отриманих даних, що дозволяє в режимі реального часу спостерігати за показниками. Для цього використано функцію *JavaScript* із циклічним викликом, яка через задані інтервали часу надсилає *AJAX*-запити до мікроконтролера та, отримавши відповідь у форматі *JSON*, оновлює вміст *HTML*-сторінки без її перезавантаження. Такий підхід забезпечує плавну зміну показників у режимі реального часу, створюючи ілюзію динамічної системи моніторингу, що значно підвищує зручність користування. Уся візуалізація даних організована у вигляді простої й зрозумілої інформаційної панелі, яка складається з окремих блоків для кожного показника. Елементи панелі стилізовано з використанням *CSS*: задано кольори фону, межі блоків, відступи, вирівнювання, розміри шрифтів і оформлення заголовків, що робить інтерфейс візуально привабливим і придатним до використання як на комп'ютерах, так і на

мобільних пристроях. Візуальне відображення кожного з показників розміщено по центру екрана, використано великі шрифти, які добре читаються з різних кутів огляду, що особливо важливо для користувачів з різним рівнем зору. Завдяки мінімалістичному підходу до дизайну забезпечено високу швидкодію вебінтерфейсу навіть за обмежених ресурсів *ESP8266*. Крім того, весь *HTML*-код інтегрується у прошивку безпосередньо у середовищі розробки *Arduino IDE* у вигляді змінної типу *String*. Це дозволяє зручно модифікувати інтерфейс та швидко впроваджувати оновлення без необхідності використовувати окремі файли чи зовнішні ресурси. На рисунку 3.6 можна побачити візуалізовані показники даної кваліфікаційної роботи.

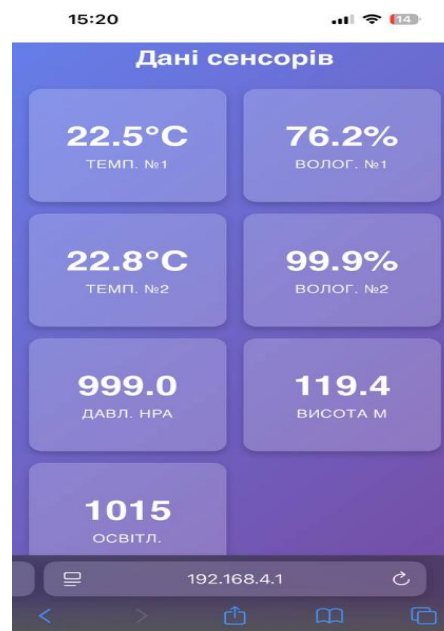


Рисунок 3.6 – Візуалізовані показники на веб-інтерфейсі

3.5 Тестування та налагодження пристрою (інструкція користувача)

У даній кваліфікаційній роботі особливу увагу було приділено етапу тестування розробленого пристрою, який поєднує функції збору, обробки та візуалізації метеорологічних показників через вбудований веб-інтерфейс, що базується на мікроконтролері *ESP8266*. У ході реалізації цього етапу проводилися всебічні дослідження працездатності пристрою в умовах реального середовища експлуатації, що передбачало всебічну перевірку

функціоналу як апаратної, так і програмної частини системи. Зокрема, особливу увагу було приділено питанню забезпечення надійності збору даних із датчиків температури, вологості, атмосферного тиску та освітленості, а також перевірці коректності відображення цих даних у вебінтерфейсі, що має ключове значення для кінцевого користувача.

Під час підготовки інструкції для кінцевого користувача було визначено основні етапи взаємодії з пристроєм та методику тестування, яка забезпечує повноцінне розкриття функціональних можливостей розробленої системи. На першому етапі користувачу рекомендується ретельно перевірити наявність належного живлення пристрою, переконатися у правильному підключенні живлення до мікроконтролера та сенсорних модулів, а також вибрати безпечне місце для розташування пристрою, що має бути добре вентильованим і захищеним від впливу зовнішніх факторів, таких як пряме сонячне світло, теплові потоки від опалювальних приладів або інші джерела тепла чи вологи, які можуть призвести до спотворення результатів вимірювань. Лише після виконання цих попередніх заходів користувач може розпочати роботу з пристроєм, що гарантує високу достовірність та точність отриманих даних.

Після підключення живлення мікроконтролер *ESP8266* автоматично ініціалізується та виконує процес підключення до заздалегідь налаштованої *Wi-Fi*-мережі, використовуючи відповідні налаштування, що були закладені у програмний код під час розробки прошивки. Завдяки використанню бібліотек *ESP8266WiFi.h* та *ESP8266WebServer.h*, у пристрої створюється внутрішній веб-сервер, який забезпечує обробку *HTTP*-запитів та надає доступ до веб-інтерфейсу з будь-якого пристрою у тій самій локальній мережі. Це рішення значно спрощує процес налаштування та взаємодії з пристроєм, роблячи його універсальним і доступним для користувачів із різним рівнем технічної підготовки.

Для початку роботи користувачу необхідно дізнатися *IP*-адресу пристрою, яка відображається у моніторі порту під час завантаження прошивки або може бути визначена через панель керування *Wi-Fi*-роутера.

Після отримання IP-адреси користувачеві достатньо увести її у адресний рядок будь-якого сучасного веб-браузера, такого як *Google Chrome*, *Mozilla Firefox* чи *Microsoft Edge*, що відкриває веб-інтерфейс пристрою. Веб-інтерфейс побудований на HTML із використанням динамічної стилізації та адаптивного дизайну, що забезпечує коректне відображення даних на пристроях із різними типами екранів та роздільною здатністю. Такий підхід гарантує універсальність та зручність використання, адже користувач може переглядати актуальні показники температури, вологості, тиску та освітленості на смартфоні, планшеті чи комп'ютері без потреби встановлення додаткового програмного забезпечення або розширень браузера.

Особливої уваги було приділено перевірці працездатності веб-інтерфейсу в умовах реального часу. Для цього у програмному коді реалізовано механізм автоматичного оновлення показників без потреби перезавантаження сторінки. Така функціональність досягається завдяки використанню *JavaScript*-коду, який із встановленою періодичністю, наприклад кожен секунду, надсилає фоновий *HTTP*-запит до спеціального маршруту */data*, реалізованого у прошивці мікроконтролера. У відповідь веб-сервер формує структуру даних у форматі *JSON*, яка містить актуальні значення сенсорів, після чого ці дані миттєво відображаються у відповідних *HTML*-елементах веб-інтерфейсу. Такий підхід забезпечує безперервний моніторинг показників та створює ілюзію «живої» роботи пристрою, що значно підвищує інформативність та зручність для користувача.

У рамках тестування працездатності пристрою проводилася перевірка стабільності підключення та оновлення даних у різних умовах: вдень і вночі, у приміщенні та на відкритому повітрі, що дозволило оцінити роботу пристрою під впливом різних температурних та вологісних режимів, а також перевірити стійкість сигналу *Wi-Fi* навіть за умов низької якості з'єднання. За результатами тестування було підтверджено високу стабільність роботи пристрою, відсутність помилок під час зчитування показників та коректну роботу веб-інтерфейсу на найпопулярніших браузерах, таких як *Google*

Chrome, Mozilla Firefox, Microsoft Edge, що доводить універсальність та сумісність системи.

Також у рамках тестування було перевірено працездатність сенсорів *BME280* та *DHT22*, а також фотодатчика освітленості, з урахуванням того, що пристрій побудований із використанням бібліотек *Adafruit_BME280.h* та *DHT.h*. Це дозволило оцінити точність та стабільність показників навіть під час зміни умов навколишнього середовища. Важливо зазначити, що у ході тестування не виникало необхідності перезапуску або перепрошивки пристрою, що підтверджує його надійність та придатність до використання у побутових або лабораторних умовах.

Для забезпечення надійної роботи системи у разі потреби перезапуску або оновлення прошивки передбачено можливість фізичного перезавантаження пристрою за допомогою кнопки, розташованої на корпусі, а також можливість оновлення прошивки через *Arduino IDE*. Це дає змогу швидко вносити необхідні зміни до логіки роботи пристрою або параметрів мережі, що особливо важливо під час розгортання пристрою у нових умовах або для оновлення його функціоналу.

У межах даної кваліфікаційної роботи також було розглянуто питання розширення функціоналу пристрою шляхом реалізації графічного режиму відображення даних у веб-інтерфейсі. Це дозволяє користувачу візуалізувати зміни показників у вигляді графіків або діаграм, що значно спрощує аналіз отриманої інформації та виявлення закономірностей. Окрім цього, передбачена можливість створення лог-файлів для збереження історії показників із подальшим їхнім аналізом, що відкриває нові можливості для використання пристрою в наукових дослідженнях або тривалому моніторингу стану навколишнього середовища.

Інструкція користувача, що є результатом даної кваліфікаційної роботи, охоплює всі необхідні аспекти взаємодії з пристроєм: підключення живлення, налаштування підключення до *Wi-Fi*, доступ до веб-інтерфейсу, інтерпретацію отриманих показників, перевірку стану сенсорів, а також дії у разі виникнення

неполадок, зокрема повторне підключення живлення, перевірку стану з'єднання або оновлення прошивки. Завдяки проведеному тестуванню вдалося підтвердити ефективність, стабільність та зручність функціонування пристрою навіть для користувачів без спеціальної технічної підготовки, що робить даний проєкт не лише навчальним, а й потенційно практичним для застосування як у побутових умовах, так і в умовах наукових досліджень. На рисунку 3.7 буде зображено макет метеостанції у зібраному варіанті.



Рисунок 3.7 – Макет метеостанції у зібраному варіанті

В результаті, кінцевий користувач отримує інтуїтивно зрозумілу та зручну у використанні платформу для моніторингу метеорологічних показників, яка поєднує у собі простоту налаштування, стабільність роботи та можливість подальшого розширення функціоналу, що є важливою перевагою для сучасних систем моніторингу. З урахуванням коду програми, який представлено у даній кваліфікаційній роботі, наведено основні елементи, зокрема бібліотеки *ESP8266WiFi.h*, *ESP8266WebServer.h*, *Wire.h*, *Adafruit_BME280.h* та *DHT.h*, що відповідають за зв'язок із мережею, обробку

веб-запитів та взаємодію з датчиками температури, вологості, тиску й освітленості.

Програмний код починається з оголошення пінів *I2C_SDA* та *I2C_SCL*, визначення пінів для сенсорів та створення об'єктів для обробки даних сенсорів. Також реалізовано функцію сканування шини *I2C* для визначення наявних підключених пристроїв, що дозволяє своєчасно виявляти несправності та підвищує надійність системи. Додатково у прошивці реалізовано обробник маршруту */data*, який забезпечує передачу актуальних значень сенсорів у форматі *JSON* для подальшого відображення у веб-інтерфейсі.

Коротка інструкція користувача. Першочергово для запуску метеостанції необхідне живлення, підходить лише кабель типу *MicroUSB – USB*. Після під'єднання живлення необхідно підключитись до мережі під назвою *ESP8266-AP*, після чого зайти у браузер і вписати *IP* адресу 192.168.4.1. Після чого ви опиняєтесь на сторінці в якій відображені всі показники, які оновлюються у реальному часі кожні 3 секунди.

Отже, у даній кваліфікаційній роботі було проведено всебічне тестування пристрою у реальних умовах, підготовлено детальну інструкцію користувача та підтверджено надійність і ефективність розробленої системи, що є одним із головних результатів дослідження та практичної цінності проєкту.

ВИСНОВКИ

У даній кваліфікаційній роботі виконано комплексне дослідження та реалізацію метеостанції з веб-інтерфейсом на базі мікроконтролера *ESP8266*. У результаті проведено аналіз предметної області, визначено основні параметри метеорологічних вимірювань, виконано порівняння існуючих рішень метеостанцій, обґрунтовано вибір апаратної та програмної частини, розроблено архітектуру системи, здійснено програмування мікроконтролера та реалізовано збір, обробку та відображення метеорологічних даних у веб-інтерфейсі. Під час роботи було обрано оптимальні датчики для вимірювання температури та атмосферного тиску, зокрема *DHT22* та *BME280*, що дозволило забезпечити високу точність вимірювань та стабільність роботи пристрою. Використання мікроконтролера *ESP8266* дало змогу реалізувати віддалений доступ до метеорологічних даних через веб-браузер, що відповідає сучасним вимогам до систем моніторингу навколишнього середовища. Розроблена система має компактні розміри, низьке енергоспоживання та достатню обчислювальну потужність, що забезпечує можливість автономної роботи в умовах обмежених ресурсів живлення. Наявність веб-інтерфейсу спрощує доступ користувачів до даних та підвищує зручність експлуатації метеостанції.

Таким чином, у процесі виконання кваліфікаційної роботи створено сучасну метеостанцію з веб-інтерфейсом на базі *ESP8266*, яка поєднує доступність, простоту та функціональність. Отримані результати демонструють можливість використання даної системи як у приватних домоволодіннях, так і для професійного моніторингу кліматичних умов. Реалізована метеостанція здатна працювати автономно, забезпечуючи збір, обробку та відображення актуальних метеорологічних параметрів, що є важливим для своєчасного реагування на зміни погодних умов та прийняття обґрунтованих рішень у різних сферах діяльності людини.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. «Adafruit BME280 Sensor». [Електронний ресурс] Режим доступу: <https://www.adafruit.com/product/2652> (дата звернення: 03.03.2025)
2. «Arduino – Open-source electronic prototyping platform». [Електронний ресурс] Режим доступу: <https://www.arduino.cc/> (дата звернення: 22.04.2025)
3. «DHT22 Digital Humidity and Temperature Sensor». [Електронний ресурс] Режим доступу: <https://www.sparkfun.com/products/10167> (дата звернення: 27.04.2025)
4. «ESP8266 Overview». [Електронний ресурс] Режим доступу до ресурсу: <https://www.espressif.com/en/products/socs/esp8266> (дата звернення: 07.04.2025)
5. «JavaScript». [Електронний ресурс] Режим доступу: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата звернення: 12.05.2025)
6. «Learn C++». [Електронний ресурс] Режим доступу: <https://www.learncpp.com/> (дата звернення: 16.05.2025)
7. «MEMS Sensors Explained». [Електронний ресурс] Режим доступу: <https://www.memsjournal.com/> (дата звернення: 16.05.2025)
8. «Microcontroller Selection Guide». [Електронний ресурс] Режим доступу: https://microchipinf.com/ua_articles/55/717 (дата звернення: 24.04.2025)
9. «MQTT Protocol Overview». [Електронний ресурс] Режим доступу: <https://mqtt.org/> (дата звернення: 08.05.2025)
10. «REST API Design». [Електронний ресурс] Режим доступу: <https://restfulapi.net/> (дата звернення: 11.05.2025)
11. Електронні метеостанції Ecowitt. [Електронний ресурс] Режим доступу: <https://www.ecowitt.com/> (дата звернення: 30.04.2025)

12. Основи метеорології. [Електронний ресурс] Режим доступу: <https://uk.wikipedia.org/wiki/Метеорологія> (дата звернення: 02.04.2025)
13. Протокол I2C. [Електронний ресурс] Режим доступу: <https://www.i2c-bus.org/> (дата звернення: 11.05.2025)
14. «Протокол SPI. [Електронний ресурс] Режим доступу: <https://www.analog.com/en/analog-dialogue/articles/introduction-to-spi-interface.html> (дата звернення: 13.05.2025)
15. «Сервер ESP8266WebServer». [Електронний ресурс] Режим доступу: <https://github.com/esp8266/ESPWebServer> (дата звернення: 20.05.2025)

ДОДАТОК А

```

#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <Wire.h>
#include <Adafruit_BME280.h>
#include <DHT.h>

// ===== ПИНЫ =====
#define I2C_SDA 4 // D2 → GPIO4
#define I2C_SCL 5 // D1 → GPIO5
#define DHT_PIN 14 // D5 → GPIO14
#define DHT_TYPE DHT22
#define LDR_PIN A0 // аналоговый вход

// ===== Объекты =====
ESP8266WebServer server(80);
Adafruit_BME280 bme;
DHT dht(DHT_PIN, DHT_TYPE);
bool foundBME = false;

// I2C-сканирование
void scanI2C() {
  Serial.println(F(">> I2C scan start"));
  for (uint8_t addr = 1; addr < 127; addr++) {
    Wire.beginTransmission(addr);
    if (Wire.endTransmission() == 0) {
      Serial.printf(" • device at 0x%02X\n", addr);
      delay(5);
    }
  }
  Serial.println(F(">> I2C scan done\n"));
}

// Обработчик JSON-данных
void handleData() {
  // Читаем
  float T_bme = foundBME ? bme.readTemperature() : NAN;
  float P_bme = foundBME ? bme.readPressure() / 100.0F : NAN;
  float H_bme = foundBME ? bme.readHumidity() : NAN;
  float A_bme = foundBME ? bme.readAltitude(1013.25F) : NAN;
  float T_dht = dht.readTemperature();
  float H_dht = dht.readHumidity();
  int LDR = analogRead(LDR_PIN);

  // Логи
  Serial.printf("→ Data: BME T=%.1f°C P=%.1f hPa H=%.1f%%
Alt=%.1fm | DHT T=%.1f°C H=%.1f%% | LDR=%d\n",
              T_bme, P_bme, H_bme, A_bme, T_dht, H_dht, LDR);
}

```

```

// Формируем JSON
String js = "{";

js += "\"T_bme\": " + String(T_bme,1) + ",";
js += "\"P_bme\": " + String(P_bme,1) + ",";
js += "\"H_bme\": " + String(H_bme,1) + ",";
js += "\"A_bme\": " + String(A_bme,1) + ",";
js += "\"T_dht\": " + String(T_dht,1) + ",";
js += "\"H_dht\": " + String(H_dht,1) + ",";
js += "\"LDR\": " + String(LDR);
js += "}";

server.send(200, "application/json; charset=utf-8", js);
}

// Обработчик главной страницы
void handleRoot() {
  Serial.println(F("→ GET /"));

  String html = R"rawliteral(
<!DOCTYPE html>
<html lang="uk">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,initial-
scale=1">
  <title>ESP8266 Dashboard</title>
  <style>
    body {
      margin:0; padding:0;
      background: linear-gradient(135deg,#667eea,#764ba2);
      color:#fff; font-family:sans-serif;
      display:flex; flex-direction:column; min-height:100vh;
    }
    header, footer { text-align:center; padding:10px; }
    header h1 { margin:0; font-size:1.6em; }
    .container {
      flex:1; display:grid;
      grid-template-columns: repeat(auto-fit,minmax(140px,1fr));
      gap:15px; padding:15px;
    }
    .card {
      background: rgba(255,255,255,0.2);
      border-radius:12px; padding:12px;
      box-shadow:0 4px 8px rgba(0,0,0,0.2);
      display:flex; flex-direction:column;
      justify-content:center; align-items:center;
    }
    .value { font-size:2.2em; font-weight:700; }
    .label { margin-top:5px; font-size:0.9em; text-
transform:uppercase; letter-spacing:0.5px; }

```

```

    footer { font-size:0.8em; }
  </style>
</head>
<body>
  <header><h1>Дані сенсорів</h1></header>
  <div class="container">
    <div class="card"><div id="t1" class="value">--.-°C</div><div
class="label">Темп. №1</div></div>
    <div class="card"><div id="h1" class="value">--.-%</div><div
class="label">Волог. №1</div></div>
    <div class="card"><div id="t2" class="value">--.-°C</div><div
class="label">Темп. №2</div></div>
    <div class="card"><div id="h2" class="value">--.-%</div><div
class="label">Волог. №2</div></div>
    <div class="card"><div id="p" class="value">----.-</div><div
class="label">Давл. hPa</div></div>
    <div class="card"><div id="alt" class="value">----.-
</div><div class="label">Висота m</div></div>
    <div class="card"><div id="ldr" class="value">----</div><div
class="label">Освітл.</div></div>
  </div>
  <footer>Автор проекту: Вячеслав Кочетов</footer>
  <script>
    async function updateData(){
      try{
        const resp = await fetch('/data');
        const js = await resp.json();
        document.getElementById('t1').innerText =
js.T_bme.toFixed(1)+'°C';
        document.getElementById('p').innerText =
js.P_bme.toFixed(1);
        document.getElementById('h1').innerText =
js.H_bme.toFixed(1)+'%';
        document.getElementById('alt').innerText=
js.A_bme.toFixed(1);
        document.getElementById('t2').innerText =
js.T_dht.toFixed(1)+'°C';
        document.getElementById('h2').innerText =
js.H_dht.toFixed(1)+'%';
        document.getElementById('ldr').innerText= js.LDR;
      } catch(e) {
        console.error('Update error', e);
      }
    }
    updateData();
    setInterval(updateData, 3000);
  </script>
</body>
</html>
)rawliteral";

  server.send(200, "text/html; charset=UTF-8", html);
}

```

```

// Обработчик 404
void handleNotFound() {
  server.send(404, "text/plain", "404: Не найдено");
}

void setup() {
  Serial.begin(115200);
  delay(100);
  Serial.println(F("\n=== ESP8266 AP Startup ==="));
  Wire.begin(I2C_SDA, I2C_SCL);
  Wire.setClock(100000);
  scanI2C();

  WiFi.softAP("ESP8266-AP");
  Serial.print(F("AP SSID: ESP8266-AP, IP: "));
  Serial.println(WiFi.softAPIP());

  // BME280 (0x76/0x77)
  for (uint8_t a : {uint8_t(0x76), uint8_t(0x77)}) {
    if (bme.begin(a)) {
      foundBME = true;
      Serial.printf("BME280 found at 0x%02X\n", a);
      break;
    }
  }
  if (!foundBME) Serial.println(F("BME280 NOT found"));

  dht.begin();
  Serial.println(F("DHT22 initialized"));

  server.on("/", handleRoot);
  server.on("/data", handleData);
  server.onNotFound(handleNotFound);
  server.begin();
  Serial.println(F("HTTP-server started\n"));
}

void loop() {
  server.handleClient();
}

```

КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ
ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА
«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»

ВІДГУК
керівника кваліфікаційної роботи

випускника спеціальності: 123 «Комп'ютерна інженерія»

відділення: комп'ютерної та програмної інженерії

циклова комісія: комп'ютерних систем та мереж

Вячеслав КОЧЕТОВ

(ім'я, прізвище)

1. Кваліфікаційна робота на тему «Розробка метеостанції на базі ESP8266» виконана здобувачем самостійно за власною ініціативою.
2. Метою роботи є проєктування та реалізація пристрою та веб-інтерфейсу для віддаленого моніторингу метеорологічних даних, зокрема температури, вологості та освітленості, у режимі реального часу.
3. Кваліфікаційна робота відповідає темі, затвердженій наказом керівництва навчального закладу.
4. Виконання проєкту здійснене здобувачем освіти самостійно з дотриманням вимог інженерії та програмування для вбудованих систем.
5. Вячеслав КОЧЕТОВ показав високий рівень володіння літературними джерелами, аналізу даних з сенсорів (BME280, DHT22, LDR), а також прийняття обґрунтованих технічних рішень і застосування сучасних IoT-технологій (ESP8266, ArduinoJson, ESPAsyncWebServer).
6. Рівень виконаної кваліфікаційної роботи заслуговує оцінку «відмінно», відповідає набутим знанням, умінням і навичкам та дозволяє можливість присвоєння йому кваліфікації фахівця освітньо-кваліфікаційного ступеню «Фаховий молодший бакалавр» спеціальності 123 «Комп'ютерна інженерія».

Керівник кваліфікаційної роботи

« » 2025 р.
(підпис)

Тетяна РУБАН
(ім'я, прізвище)

КРИВОРІЗЬКИЙ ФАХОВИЙ КОЛЕДЖ
ДЕРЖАВНОГО НЕКОМЕРЦІЙНОГО ПІДПРИЄМСТВА
«ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»

РЕЦЕНЗІЯ
на кваліфікаційну роботу

випускника спеціальності: 123 «Комп'ютерна інженерія»
відділення: комп'ютерної та програмної інженерії
циклова комісія: комп'ютерних систем та мереж
Вячеслава КОЧЕТОВА
(ім'я, прізвище)

Тематика розробки автоматизованих систем моніторингу докілья є вкрай актуальною у зв'язку з поширенням концепції "розумного дому", потребами в локальному контролі метеоумов для аграрного сектору, екологічного моніторингу та наукових досліджень. Вибір мікроконтролера ESP8266 як основи для системи є обґрунтованим завдяки його широким можливостям для IoT-проектів.

Кваліфікаційна робота на тему «Метеостанція з веб-інтерфейсом на базі ESP» повністю відповідає затвердженій темі. Усі поставлені завдання були успішно реалізовані: проведено моделювання системи, здійснено вибір апаратних компонентів, реалізовано програмне забезпечення для збору та передачі даних через веб-інтерфейс.

В результаті виконання роботи створено функціональну метеостанцію, здатну в режимі реального часу вимірювати та відображати основні параметри навколишнього середовища: температуру, вологість, тиск, освітленість та висоту. Реалізований веб-інтерфейс забезпечує зручний доступ до даних для кінцевого користувача.

Пояснювальна записка оформлена відповідно до вимог ДСТУ. Робота структурована логічно, з дотриманням наукового стилю. Окремо слід відзначити якісне подання схем, алгоритмів та програмних рішень, що свідчить про належний рівень практичної підготовки автора.

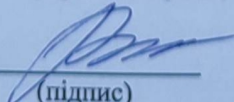
Розроблений пристрій має значний потенціал практичного застосування. Може бути використаний як в освітніх цілях, так і для реального моніторингу в домашніх чи польових умовах. Можлива інтеграція з іншими системами, що розширює функціонал розробки.

Кваліфікаційна робота виконана на високому рівні, демонструє знання мікроконтролерного програмування, роботи з периферійними модулями та базові навички веб-розробки.

Робота заслуговує на оцінку "відмінно".


Рецензент _____ викладач _____
(науковий ступінь, посада)

« 30 » 05 2025 р.


(підпис)

Сергій РУДИЙ
(ім'я, прізвище)

З рецензією ознайомлений


(підпис)

Вячеслав КОЧЕТОВ
(ім'я, прізвище)