

Міністерство освіти і науки України

Відокремлений структурний підрозділ «Тернопільський фаховий коледж
Тернопільського національного технічного університету імені Івана Пулюя»

(повне найменування вищого навчального закладу)

Відділення інформаційних технологій, менеджменту, туризму
та підготовки іноземних громадян

(назва відділення)

Циклова комісія комп'ютерної інженерії

(повна назва циклової комісії)

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи

фахового молодшого бакалавра

(освітньо-професійного ступеня)

на тему: Розробка програмного забезпечення IoT-системи моніторингу якості
повітря та керування вентиляцією на базі ESP32

Виконав: студент IV курсу, групи KI-405п

Спеціальності 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

Владислав ВІВЧАР
(ім'я та прізвище)

Керівник Роксолана БОЛЮБАШ
(ім'я та прізвище)

Рецензент _____
(ім'я та прізвище)

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ТЕРНОПІЛЬСЬКИЙ ФАХОВИЙ КОЛЕДЖ
ТЕРНОПІЛЬСЬКОГО НАЦІОНАЛЬНОГО ТЕХНІЧНОГО
УНІВЕРСИТЕТУ
імені ІВАНА ПУЛЮЯ»**

Відділення інформаційних технологій, менеджменту, туризму
та підготовки іноземних громадян

Циклова комісія комп'ютерної інженерії

Освітньо-професійний ступінь фаховий молодший бакалавр

Освітньо-професійна програма: Обслуговування комп'ютерних систем і мереж

Спеціальність: 123 Комп'ютерна інженерія

Галузь знань: 12 Інформаційні технології

ЗАТВЕРДЖУЮ

Голова циклової комісії
комп'ютерної інженерії

_____ Андрій ЮЗЬКІВ

“30” березня 2026 року

**З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Вівчару Владиславу Володимировичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32

керівник роботи Болюбаш Роксолана Юріївна
(прізвище, ім'я, по батькові)

затверджені наказом Відокремленого структурного підрозділу «Тернопільський фаховий коледж Тернопільського національного технічного університету імені Івана Пулюя» від 27.03.2026р № 4/9-167.

2. Строк подання студентом роботи: 15 червня 2026 року.

3. Вихідні дані до роботи: методичні вказівки для виконання кваліфікаційної роботи, середовище Arduino, C++

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): Загальний розділ. Розробка технічного та робочого проєкту. Спеціальний розділ. Економічний розділ. Охорона праці та безпека життєдіяльності.

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

- структурна схема IoT-системи;
- блок-схема роботи програмного забезпечення;
- діаграма розгортання (Deployment Diagram);
- таблиця техніко-економічних показників для пристрою обслуговування.

6. Консультанти розділів роботи

Розділ	Ім'я, прізвище та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний розділ	Богдана МАРТИНЮК викладач		
Охорона праці та безпека життєдіяльності	Володимир ШТОКАЛО викладач		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Отримання і аналіз технічного завдання	01.04	
2	Збір і узагальнення інформації	08.05	
3	Написання першого розділу	15.05	
4	Розробка технічного та робочого проекту	22.05	
5	Написання спеціального розділу	28.05	
6	Розрахунок економічної частини	1.06	
7	Написання розділу охорони праці	3.06	
8	Виконання графічної частини	8.06	
9	Оформлення проекту	10.06	
10	Погодження нормоконтролю	11.06	
11	Попередній захист роботи	12.06	
12	Захист кваліфікаційної роботи		

7. Дата видачі завдання: 31 березня 2026 року

Студент

(підпис)

Керівник роботи

(підпис)

Владислав ВІВЧАР

(ім'я та прізвище)

Роксолана БОЛЮБАШ

(ім'я та прізвище)

ЗМІСТ

ВСТУП.....	8
1 ЗАГАЛЬНИЙ РОЗДІЛ.....	9
1.1 Аналітичний огляд існуючих рішень.....	9
1.2 Технічне завдання.....	10
1.2.1 Найменування та область застосування.....	10
1.2.2 Призначення розробки.....	11
1.2.3 Вимоги до апаратного та програмного забезпечення.....	12
1.2.4 Вимоги до документації.....	13
1.2.5 Техніко-економічні показники.....	14
1.2.6 Стадії та етапи розробки.....	15
1.2.7 Порядок контролю та прийому.....	17
2 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЄКТУ.....	19
2.1 Постановка задачі на розробку програмного забезпечення.....	19
2.2 Опис та обґрунтування вибору структури та методу організації вхідних та вихідних даних.....	22
2.3 Розробка алгоритму.....	25
2.3.1 Зовнішнє проєктування програми.....	25
2.3.2 Проєктування логіки програми.....	29
2.4 Визначення інформаційних зв'язків.....	33
2.5 Написання текстів програм.....	36
2.6 Тестування та налагодження програм.....	39
3 СПЕЦІАЛЬНИЙ РОЗДІЛ.....	40

					<i>2026.KBP.123.405.04.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	Розробка програмного забезпечення IoT-системи моніторингу повітря та керування вентиляцією на базі ESP32 Пояснювальна записка	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Розроб.</i>		Вівчар В.В.						
<i>Перевір.</i>							5	76
<i>Реценз.</i>						ВСП «ТФК ТНТУ» ТНТУ		
<i>Н. Контр.</i>								

3.1 Інструкція з інсталяції та налаштування вебсерверів та супутнього програмного забезпечення	44
3.2 Інструкція по встановленню та налаштуванню розробленого проєкту	47
3.3 Інструкція з використання тестових наборів	45
3.4 Інструкція з експлуатації програмного комплексу	46
4 ЕКОНОМІЧНИЙ РОЗДІЛ	47
4.1 Визначення стадій технологічного процесу та загальної тривалості проведення НДР	47
4.2 Розрахунок витрат на оплату праці	48
4.3 Розрахунок відрахувань на соціальні заходи	50
4.4 Розрахунок суми амортизаційних відрахувань	52
4.5 Розрахунок накладних витрат... ..	52
4.6 Складання кошторису витрат та визначення собівартості НДР... ..	54
4.7 Визначення ціни НДР	55
4.8 Визначення економічної ефективності і терміну окупності капітальних вкладень	56
5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ	58
5.1 Вплив параметрів мікроклімату на організм людини	58
5.1.1 Температура повітря та її вплив	58
5.1.2 Відносна вологість повітря	59
5.1.3 Якість повітря та газові домішки	60
5.1.4 Рух повітря та вентиляція	62
5.2 Вентиляція виробничих приміщень	62
5.3 Організація безпечної експлуатації низьковольтних інтелектуальних систем	65
5.3.1 Електробезпека при експлуатації IoT-систем	66
5.3.2 Захист апаратної частини системи	66
5.3.3 Програмна безпека та надійність роботи	67

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						6
Змі	Арк	№ докум.	Підпис	Дата		

ВИСНОВКИ68

ПЕРЕЛІК ПОСИЛАНЬ.....69

ДОДАТКИ 71

ВСТУП

Стрімкий розвиток технологій Інтернету речей (IoT) сприяє створенню інтелектуальних систем моніторингу навколишнього середовища. Одним із важливих напрямів їх застосування є контроль параметрів повітря в приміщеннях, оскільки від якості повітря залежать комфорт, працездатність та здоров'я людей.

Актуальність теми полягає в необхідності створення доступних систем моніторингу повітря, здатних у режимі реального часу збирати та передавати інформацію про стан повітряного середовища. Використання мікроконтролерів сімейства ESP32 дозволяє реалізувати такі системи з мінімальними витратами та високою ефективністю.

Метою кваліфікаційної роботи є розробка програмного забезпечення IoT-системи моніторингу повітря та керування вентиляцією на базі ESP32.

Об'єктом дослідження є процес моніторингу параметрів повітряного середовища із застосуванням технологій Інтернету речей.

Предметом дослідження є програмне забезпечення IoT-системи моніторингу повітря на базі ESP32.

Практичне значення роботи полягає у створенні програмно-апаратного комплексу, який забезпечує контроль температури, вологості та якості повітря з можливістю локального і віддаленого моніторингу через платформу Blynk IoT.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						8
Змі	Арк	№ докум.	Підпис	Дата		

1 ЗАГАЛЬНИЙ РОЗДІЛ

1.1 Аналітичний огляд існуючих рішень

У сучасних умовах значна увага приділяється контролю параметрів навколишнього середовища, зокрема якості повітря в житлових, офісних та виробничих приміщеннях. Для вирішення цієї задачі широко застосовуються системи моніторингу повітря, які дозволяють контролювати температуру, вологість та рівень забруднення повітря в режимі реального часу.

На ринку представлена велика кількість готових рішень для моніторингу якості повітря. До них належать як автономні побутові пристрої, так і комплексні системи на базі технологій Інтернету речей (IoT). Більшість сучасних систем використовують датчики температури, вологості, концентрації вуглекислого газу та інших шкідливих речовин, а також забезпечують віддалений доступ до результатів вимірювань через мобільні додатки або веб-інтерфейси.

Одним із популярних напрямків є використання мікроконтролерів сімейства ESP32, які поєднують достатню обчислювальну потужність, вбудовані модулі бездротового зв'язку Wi-Fi та низьку вартість. Завдяки цьому вони широко застосовуються при створенні IoT-систем моніторингу навколишнього середовища. Для передачі та візуалізації даних часто використовуються хмарні платформи, серед яких значного поширення набула платформа Blynk IoT.

Аналіз існуючих рішень показав, що більшість комерційних систем мають високу вартість або обмежені можливості налаштування під потреби користувача. Крім того, деякі системи використовують закриті програмні рішення, що ускладнює їх модернізацію та розширення функціональних можливостей.

З огляду на це доцільним є розроблення власної IoT-системи моніторингу повітря на базі мікроконтролера ESP32 із використанням датчиків DHT11 та MQ135. Такий підхід забезпечує можливість контролю температури, вологості та якості повітря, локального відображення результатів на LCD-дисплеї та віддаленого моніторингу за допомогою платформи Blynk IoT.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						9
Змн.	Арк	№ докум.	Підпис	Дата		

1.2 Технічне завдання

1.2.1 Найменування та область застосування

Найменування розробки – «Програмне забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32».

Розроблюване програмне забезпечення призначене для автоматизованого контролю параметрів якості повітря в приміщеннях та керування вентиляційним обладнанням на основі отриманих даних. Система забезпечує збір інформації з підключених датчиків, її обробку, передачу мережею Wi-Fi та формування керуючих сигналів для вентиляційної системи.

Областю застосування розробки є житлові, офісні, навчальні та виробничі приміщення, у яких необхідно здійснювати постійний моніторинг параметрів повітряного середовища та автоматично підтримувати комфортні умови перебування людей.

Програмне забезпечення функціонує на базі мікроконтролера ESP32 та взаємодіє з датчиками контролю якості повітря, зокрема датчиком MQ-135, а також виконавчими пристроями системи вентиляції через релейні модулі. Передача даних здійснюється бездротовими каналами зв'язку з використанням мережі Wi-Fi. Розроблена система може використовуватися як самостійне рішення для контролю якості повітря або бути інтегрованою до складу систем автоматизації будівель та концепції «розумного будинку» (Smart Home). Основними користувачами системи є власники приміщень, обслуговуючий персонал та оператори систем моніторингу навколишнього середовища.

Застосування розробленого програмного забезпечення дозволяє підвищити ефективність контролю параметрів повітря, своєчасно реагувати на погіршення його якості та зменшити витрати на ручне керування вентиляційним обладнанням.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						10
Змн.	Арк	№ докум.	Підпис	Дата		

1.2.2 Призначення розробки

Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 призначена для автоматизації процесів контролю стану повітряного середовища у приміщеннях та керування вентиляційним обладнанням відповідно до отриманих показників.

Основним призначенням системи є безперервний збір даних з датчиків якості повітря, аналіз отриманої інформації та прийняття рішень щодо вмикання або вимикання вентиляції без безпосереднього втручання користувача. Це дозволяє підтримувати комфортні та безпечні умови перебування людей у приміщенні, а також забезпечувати раціональне використання електроенергії.

Програмне забезпечення забезпечує виконання таких функцій:

- зчитування показників датчика якості повітря MQ-135;
- обробку та аналіз отриманих даних;
- відображення поточного стану системи на LCD-дисплеї;
- передачу даних через бездротову мережу Wi-Fi;
- автоматичне керування вентиляцією за допомогою релейного модуля;
- збереження та оновлення інформації про параметри повітря;

інформування користувача про перевищення встановлених порогових значень забруднення повітря.

Розроблювана система орієнтована на використання у житлових будинках, навчальних закладах, офісних приміщеннях та інших об'єктах, де необхідно здійснювати постійний контроль якості повітря та забезпечувати своєчасне провітрювання приміщень.

Використання системи дозволяє підвищити ефективність моніторингу параметрів навколишнього середовища, зменшити вплив людського фактора на процес керування вентиляцією та забезпечити більш комфортні умови експлуатації приміщень.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						11
Змн.	Арк	№ докум.	Підпис	Дата		

1.2.3 Вимоги до апаратного та програмного забезпечення

Для забезпечення коректного функціонування IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 необхідно використовувати відповідне апаратне та програмне забезпечення.

До складу апаратної частини системи повинні входити:

- мікроконтролер ESP32 з підтримкою бездротового зв'язку Wi-Fi;
- датчик якості повітря MQ-135 для контролю концентрації шкідливих речовин у повітрі;
- LCD-дисплей 1602 для відображення поточних параметрів системи;
- релейний модуль для керування вентиляційним обладнанням;
- вентилятор або інший виконавчий пристрій вентиляційної системи;
- джерело живлення напругою 5 В;
- з'єднувальні проводи та допоміжні електронні компоненти.

Для розробки, тестування та налагодження програмного забезпечення необхідно використовувати персональний комп'ютер з такими мінімальними характеристиками:

- процесор із тактовою частотою не менше 2,0 ГГц;
- оперативна пам'ять обсягом не менше 4 ГБ;
- вільний простір на накопичувачі не менше 10 ГБ;
- наявність USB-порту для програмування мікроконтролера;
- підключення до мережі Інтернет.

Для розробки та функціонування системи необхідно використовувати таке програмне забезпечення:

- операційну систему Windows 10 або новішу;
- середовище розробки Arduino IDE;
- драйвери для програмування та роботи з платою ESP32;
- бібліотеки для роботи з датчиком MQ-135, LCD-дисплеєм та мережевими інтерфейсами;

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						12
Змн.	Арк	№ докум.	Підпис	Дата		

- програмні засоби для налагодження та моніторингу роботи системи. Програмне забезпечення мікроконтролера повинно забезпечувати:
- періодичне зчитування даних із датчика MQ-135;
- обробку та аналіз отриманих показників;
- відображення інформації на LCD-дисплеї;
- передачу даних через мережу Wi-Fi;
- автоматичне керування вентиляцією залежно від рівня забруднення повітря;
- стабільну роботу системи в режимі безперервного моніторингу;
- можливість подальшого розширення функціоналу та інтеграції з іншими IoT-пристроями.

Апаратне та програмне забезпечення повинні забезпечувати надійне функціонування системи, своєчасне реагування на зміну параметрів якості повітря та безпечне керування вентиляційним обладнанням.

1.2.4 Вимоги до документації

У процесі розробки програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 повинна бути підготовлена комплектна документація, необхідна для впровадження, експлуатації, обслуговування та подальшої модернізації системи.

Документація повинна бути оформлена відповідно до чинних державних стандартів України та вимог, що висуваються до дипломних проєктів зі спеціальності, із дотриманням установлених правил оформлення текстових документів.

До складу документації повинні входити:

- пояснювальна записка до дипломного проєкту;
- технічне завдання на розробку програмного забезпечення;
- опис структури та принципу роботи системи;
- опис апаратної частини системи;

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						13
Змн.	Арк	№ докум.	Підпис	Дата		

- опис програмного забезпечення та алгоритмів його роботи;
- схема підключення компонентів системи;
- інструкція користувача;
- інструкція з налаштування та запуску системи;
- результати тестування та перевірки працездатності програмного забезпечення;
- графічні матеріали та додатки, необхідні для пояснення принципу роботи системи.

Пояснювальна записка повинна містити обґрунтування актуальності розробки, аналіз існуючих рішень, опис архітектури системи, характеристику використаних апаратних та програмних засобів, результати розробки, економічне обґрунтування та розділ з охорони праці.

Документація повинна забезпечувати можливість самостійного встановлення, налаштування та експлуатації системи користувачем без необхідності залучення розробника. Усі технічні рішення, алгоритми роботи та особливості функціонування програмного забезпечення повинні бути описані достатньо детально для подальшого супроводу та модернізації системи.

Оформлення документації повинно виконуватися українською мовою з використанням загальноприйнятої технічної термінології та відповідати вимогам нормативних документів щодо оформлення технічної документації.

1.2.5 Техніко-економічні показники

Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 спрямована на підвищення ефективності контролю параметрів повітряного середовища та автоматизацію процесу керування вентиляційним обладнанням. Використання системи дозволяє зменшити вплив людського фактора, підвищити оперативність реагування на зміни стану повітря та забезпечити комфортні умови перебування людей у приміщенні

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						14
Змн.	Арк	№ докум.	Підпис	Дата		

Основними технічними перевагами розробки є:

- автоматичний моніторинг показників якості повітря в режимі реального часу;
- бездротова передача даних за допомогою технології Wi-Fi;
- автоматичне керування вентиляцією відповідно до встановлених порогових значень;
- можливість цілодобової роботи без участі оператора;
- простота масштабування та інтеграції з іншими IoT-пристроями;
- низька вартість апаратної реалізації завдяки використанню мікроконтролера ESP32.

До основних техніко-економічних показників проєкту належать:

- загальна тривалість виконання науково-дослідної роботи – 186 годин;
- собівартість розробки – 142 648,30 грн;
- договірна ціна НДР – 222 531,35 грн;
- плановий прибуток – 79 883,05 грн;
- термін окупності проєкту – 1,79 року.

Впровадження розробленої системи дозволяє знизити витрати часу на контроль якості повітря, забезпечити своєчасне ввімкнення вентиляції при погіршенні параметрів повітряного середовища та підвищити ефективність використання вентиляційного обладнання. Автоматизація процесів моніторингу та керування сприяє зменшенню експлуатаційних витрат і покращенню умов праці та перебування людей у приміщеннях.

Таким чином, розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 є технічно доцільною та економічно обґрунтованою, а її впровадження забезпечує отримання позитивного економічного ефекту та підвищення рівня автоматизації об'єктів.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						15
Змн.	Арк	№ докум.	Підпис	Дата		

1.2.6 Стадії та етапи розробки

Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 виконується поетапно, що дає змогу послідовно перейти від формування вимог до отримання працездатного програмного продукту. Поділ на стадії забезпечує контроль якості робіт, зменшує ризик помилок та дозволяє своєчасно перевіряти правильність прийнятих технічних рішень.

На першій стадії виконується аналіз предметної області та формування вимог. На цьому етапі визначаються функції системи, склад апаратних засобів, перелік датчиків, принцип взаємодії між мікроконтролером ESP32, виконавчими пристроями та користувачем. Також уточнюються показники, які необхідно контролювати: температура, вологість, концентрація шкідливих газів та інші параметри повітря. Окремо встановлюються вимоги до автоматичного керування вентиляцією залежно від отриманих даних.

Друга стадія — проектування програмної та логічної структури системи. На цьому етапі розробляється загальна архітектура IoT-рішення, визначаються модулі програми, алгоритми збору та обробки даних, логіка прийняття рішень щодо вмикання або вимикання вентиляції. Паралельно обираються способи передачі даних, формат їх зберігання та варіанти відображення інформації користувачеві.

Третя стадія — розробка програмного забезпечення. Вона включає написання коду для ESP32, реалізацію опитування сенсорів, обробку показників, керування реле або іншими виконавчими елементами, а також організацію обміну даними з інтерфейсом моніторингу. На цьому етапі створюються основні програмні модулі, що забезпечують роботу системи в автоматичному режимі.

Четверта стадія — налагодження та тестування. Після реалізації програмного коду перевіряється правильність роботи окремих функцій і системи загалом. Тестування охоплює контроль зчитування даних із датчиків, коректність передачі інформації, реакцію системи на зміни параметрів повітря та надійність керування вентиляцією. За результатами випробувань виявлені недоліки усуваються, а

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						16
Змн.	Арк	№ докум.	Підпис	Дата		

алгоритми за потреби уточнюються.

П'ята стадія — інтеграція та перевірка в складі готової системи. Програмне забезпечення випробовується разом з апаратною частиною, після чого оцінюється стабільність роботи IoT-системи в умовах, наближених до реальних. На цьому етапі підтверджується працездатність усіх вузлів: датчиків, контролера ESP32, системи зв'язку та вентиляційного керування.

Завершальною стадією є оформлення документації. Підготовляються пояснювальні матеріали щодо структури програми, її функцій, алгоритмів роботи та правил експлуатації. Документація потрібна для подальшого супроводу, модернізації та можливого розширення функціональності системи.

Отже, стадії та етапи розробки програмного забезпечення IoT-системи охоплюють аналіз вимог, проектування, реалізацію, тестування, інтеграцію та документування. Така послідовність забезпечує створення надійного програмного рішення для моніторингу якості повітря і автоматичного керування вентиляцією.

1.2.7 Порядок контролю та прийому

Контроль і приймання програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 виконуються з метою перевірки відповідності розробленого рішення встановленим вимогам, функціональному призначенню та умовам подальшої експлуатації. Даний етап є завершальним у процесі розробки та дає змогу переконатися, що програмний продукт працює стабільно, коректно обробляє дані з датчиків і забезпечує правильне керування виконавчими пристроями.

Контроль якості програмного забезпечення здійснюється послідовно на кількох рівнях. Спочатку виконується внутрішній контроль окремих програмних модулів, під час якого перевіряється правильність роботи функцій зчитування даних, обробки показників, передачі інформації та керування реле. На цьому етапі основна увага приділяється виявленню синтаксичних помилок, логічних неточностей і некоректної взаємодії між окремими частинами програми.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						17
Змн.	Арк	№ докум.	Підпис	Дата		

Після цього проводиться функціональне тестування. Воно передбачає перевірку основних режимів роботи системи, зокрема: отримання даних із сенсорів, відображення поточних значень, реакцію програми на перевищення допустимих порогів якості повітря та вмикання або вимикання вентиляції залежно від заданого алгоритму. Під час такого тестування оцінюється правильність виконання всіх передбачених функцій у реальних або імітованих умовах.

Окремо здійснюється перевірка інтеграції програмного забезпечення з апаратною частиною. На цьому етапі контролюється взаємодія ESP32 із датчиками якості повітря, модулем реле, дисплеєм та іншими елементами системи. Важливо переконатися, що передача сигналів відбувається без затримок і збоїв, а виконавчі пристрої реагують на команди відповідно до алгоритму керування.

Додатково проводиться перевірка надійності та стійкості роботи. Програмне забезпечення має коректно функціонувати протягом тривалого часу без зависань, помилкового спрацювання або втрати даних. У разі виявлення відхилень виконуються доопрацювання програмного коду та повторне тестування. Також може перевірятися поведінка системи при тимчасовій відсутності даних із датчиків, перезавантаження мікроконтролера або виникненні інших нештатних ситуацій. Приймання розробленого програмного забезпечення здійснюється після завершення всіх перевірок і усунення виявлених недоліків. Під час приймання оцінюється відповідність системи технічному завданню, повнота реалізованих функцій, зручність використання та готовність до експлуатації. Основними критеріями приймання є коректне зчитування показників якості повітря, своєчасне спрацювання системи вентиляції, стабільний обмін даними та можливість безперебійної роботи в заданому режимі.

За результатами контролю складається висновок про придатність програмного забезпечення до використання. Якщо система відповідає всім вимогам, вона вважається прийнятною та може бути рекомендована до подальшої експлуатації або впровадження. У разі невідповідності окремих параметрів проводиться додаткове коригування алгоритмів і повторна перевірка.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						18
Змн.	Арк	№ докум.	Підпис	Дата		

Таким чином, порядок контролю та приймання включає поетапну перевірку програмних модулів, функціональне тестування, інтеграційні випробування, оцінку надійності та остаточне приймання системи. Це забезпечує якість розробки та підтверджує працездатність програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією.

1.2 Постановка задачі на розробку проекту. Характеристика підприємства, для якого створюється проект мережі

Проект розробки програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 спрямований на створення автоматизованого рішення для контролю параметрів повітряного середовища та забезпечення комфортних і безпечних умов перебування людей у приміщеннях. Основною задачею є розробка програмного забезпечення, яке дозволяє здійснювати безперервний моніторинг стану повітря та автоматичне керування вентиляційними пристроями залежно від отриманих показників.

Система повинна забезпечувати збір даних із датчиків, що вимірюють температуру, вологість, а також концентрацію шкідливих газів і домішок у повітрі. Отримана інформація має оброблятися в режимі реального часу мікроконтролером ESP32 з подальшим прийняттям рішень щодо вмикання або вимикання вентиляції. Додатково система повинна передбачати можливість передачі даних для моніторингу, а також відображення поточних параметрів користувачу.

До основних задач проекту належать: розробка алгоритмів зчитування даних із сенсорів, реалізація логіки обробки отриманих показників, створення механізму керування виконавчими пристроями (реле, вентилятори), а також забезпечення стабільної роботи системи в автономному режимі. Важливою вимогою є також енергоефективність, надійність і здатність системи працювати без постійного втручання користувача.

Розроблювана система орієнтована на використання в умовах малих та середніх підприємств, офісних приміщень, навчальних аудиторій або

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						19
Змн.	Арк	№ докум.	Підпис	Дата		

лабораторій, де необхідний постійний контроль якості повітря та автоматичне регулювання вентиляції. Як умовне підприємство для впровадження системи можна розглядати освітній заклад або адміністративну установу, що має декілька приміщень із різним рівнем заповненості людьми.

Такі підприємства характеризуються наступними особливостями: наявність закритих приміщень з обмеженою природною вентиляцією, змінна інтенсивність перебування людей, а також необхідність дотримання санітарно-гігієнічних норм щодо якості повітря. У подібних умовах актуальним є автоматизований контроль параметрів повітряного середовища, оскільки ручне регулювання вентиляції не завжди є ефективним або своєчасним.

Використання IoT-системи на базі ESP32 дозволяє підприємству знизити витрати на енергоспоживання вентиляційних систем, підвищити комфорт перебування у приміщеннях та забезпечити оперативне реагування на зміни показників якості повітря. Крім того, така система може бути масштабована для використання у великій кількості приміщень з централізованим моніторингом.

Отже, розробка даного проекту спрямована на підвищення ефективності управління мікрокліматом у приміщеннях підприємства та автоматизацію процесів контролю якості повітря.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						20
Змн.	Арк	№ докум.	Підпис	Дата		

2 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЄКТУ

2.1 Постановка задачі на розробку програмного забезпечення

У межах кваліфікаційної роботи виконано розробку програмного забезпечення IoT-системи моніторингу повітря та керування вентиляцією на базі мікроконтролерної платформи ESP32. Необхідність створення такої системи обумовлена потребою у постійному контролі параметрів мікроклімату приміщення та своєчасному виявленні погіршення якості повітря. Використання технологій Інтернету речей дозволяє організувати віддалений моніторинг параметрів середовища та оперативне інформування користувача про небезпечні відхилення від норми.

Об'єктом автоматизації є процес контролю параметрів повітряного середовища в приміщенні. Контроль здійснюється за такими показниками, як температура повітря, відносна вологість та рівень забруднення повітря. Для отримання зазначених параметрів використано датчик температури і вологості DHT11 та датчик якості повітря MQ135.

Технічною основою системи є мікроконтролерна плата XIAO ESP32C6, яка забезпечує обробку даних, взаємодію з периферійними пристроями та передачу інформації через мережу Wi-Fi. Для локального відображення результатів вимірювань використовується рідкокристалічний дисплей LCD 16×2 з інтерфейсом I2C, а для віддаленого моніторингу та сповіщення користувача застосовується хмарна платформа Blynk IoT.

Основним призначенням програмного забезпечення є автоматизований збір даних із датчиків, аналіз отриманої інформації, візуалізація результатів на локальному дисплеї та передача показників до хмарного сервісу. Додатково програмне забезпечення повинно забезпечувати виявлення випадків погіршення якості повітря та формування відповідних повідомлень для користувача.

Вхідними даними програмного забезпечення є:

- ☐ значення температури повітря, отримані з датчика DHT11;

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						21
Змн.	Арк	№ докум.	Підпис	Дата		

- значення відносної вологості повітря, отримані з датчика DHT11;
- значення концентрації газів та рівня забруднення повітря, отримані з датчика MQ135;
- параметри підключення до мережі Wi-Fi;
- налаштування платформи Blynk IoT.

Вихідними даними системи є:

- поточні значення температури повітря;
- поточні значення відносної вологості;
- поточні показники якості повітря;
- повідомлення про стан повітряного середовища;
- інформація для відображення на LCD-дисплеї;
- дані, передані до хмарної платформи Blynk IoT;
- подія `pollution_alert` у разі перевищення порогового значення.

Для формалізації задачі використовуються такі математичні співвідношення.

$T = \text{readTemperature}(\text{DHT11})$

$H = \text{readHumidity}(\text{DHT11})$

$AQ = \text{analogRead}(\text{MQ135})$

$S = 0$, якщо $AQ < 1200$; $S = 1$, якщо $AQ \geq 1200$

Де T — температура повітря, H — відносна вологість, AQ — цифрове значення датчика MQ135, S — ознака стану повітря: 0 відповідає задовільному стану, 1 — забрудненому стану.

У процесі розробки програмного забезпечення було поставлено такі основні завдання:

- Реалізувати підключення мікроконтролера XIAO ESP32C6 до бездротової мережі Wi-Fi.
- Забезпечити зчитування даних із датчика температури та вологості DHT11.
- Реалізувати зчитування показників датчика якості повітря MQ135.
- Виконати обробку та аналіз отриманих даних.
- Організувати локальне відображення результатів вимірювань на LCD-дисплеї.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						22
Змн.	Арк	№ докум.	Підпис	Дата		

- Реалізувати передачу даних до платформи Blynk IoT у режимі реального часу.
- Забезпечити відображення параметрів системи на вебпанелі та в мобільному застосунку Blynk.
- Реалізувати механізм автоматичного формування повідомлень про погіршення якості повітря.

Забезпечити безперервне функціонування системи в режимі моніторингу. До програмного забезпечення висуваються такі вимоги:

- коректне зчитування та обробка показників датчиків;
- стабільне підключення до мережі Wi-Fi;
- своєчасне оновлення даних на платформі Blynk IoT;
- наочне відображення інформації на LCD-дисплеї;
- формування повідомлень при перевищенні встановленого порогового значення забруднення повітря;
- надійна робота системи в режимі тривалого безперервного функціонування.

Результатом виконання поставленої задачі є програмне забезпечення IoT-системи моніторингу повітря, яке забезпечує автоматичний контроль параметрів мікроклімату приміщення, локальне та віддалене відображення результатів вимірювань, а також своєчасне інформування користувача про погіршення якості повітря. Розроблена система може використовуватися в житлових, навчальних, офісних та виробничих приміщеннях для підвищення ефективності контролю стану повітряного середовища.

2.2 Опис та обґрунтування вибору структури та методу організації вхідних та вихідних даних

Ефективність роботи IoT-системи моніторингу повітря значною мірою залежить від правильного вибору структури та способу організації вхідних і вихідних даних. Від цього залежить швидкість обробки інформації, стабільність функціонування програмного забезпечення, можливість віддаленого моніторингу та своєчасне реагування на зміни параметрів навколишнього середовища.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						23
Змн.	Арк	№ докум.	Підпис	Дата		

У розробленій системі вхідні дані формуються за рахунок показників датчиків DHT11 та MQ135, а також службової інформації, необхідної для забезпечення роботи мережесервісів. Обробка даних виконується мікроконтролером XIAO ESP32C6, після чого результати передаються на LCD-дисплей та до хмарної платформи Blynk IoT.

До складу вхідних даних належать:

- значення температури повітря, отримані від датчика DHT11;
- значення відносної вологості повітря, отримані від датчика DHT11;
- значення рівня забруднення повітря, отримані від датчика MQ135;
- параметри мережевого підключення Wi-Fi;
- ідентифікаційні параметри платформи Blynk IoT.

Датчик DHT11 формує цифрові дані, що містять інформацію про температуру та відносну вологість повітря. Отримані значення надходять до програмного забезпечення через цифровий інтерфейс введення-виведення мікроконтролера. Газовий датчик MQ135 формує аналоговий сигнал, величина якого залежить від концентрації шкідливих домішок у повітрі. Для обробки цього сигналу використовується вбудований аналого-цифровий перетворювач мікроконтролера ESP32.

Структура вхідних даних організована таким чином, щоб кожен тип інформації оброблявся окремим програмним модулем. Це забезпечує незалежність роботи окремих компонентів системи та спрощує подальшу модернізацію програмного забезпечення. Дані температури та вологості зчитуються за допомогою функцій бібліотеки DHT, а значення якості повітря отримуються через функцію аналогового зчитування `analogRead()`.

Після отримання всі вхідні дані проходять етап перевірки та попередньої обробки. Для датчика DHT11 здійснюється контроль коректності отриманих значень. У випадку виникнення помилки вимірювання дані не використовуються для подальшої обробки, а система продовжує роботу до наступного циклу опитування. Значення датчика MQ135 аналізуються шляхом порівняння з установленим пороговим значенням, що дозволяє визначати стан повітряного сер.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						24
Змн.	Арк	№ докум.	Підпис	Дата		

Вихідні дані системи формуються після завершення обробки вимірних параметрів. До них належать:

- значення температури повітря;
- значення відносної вологості повітря;
- показники якості повітря;
- повідомлення про перевищення допустимого рівня забруднення;
- інформація для локального відображення на LCD-дисплеї;
- дані для передачі до хмарної платформи Blynk IoT.

Для організації вихідних даних використано комбінований підхід, який поєднує локальне та віддалене відображення інформації. Локальне відображення реалізовано за допомогою LCD-дисплея 16×2, підключеного через інтерфейс I2C. На дисплей по чергові виводяться значення температури, вологості та оцінка якості повітря.

Таблиця 2.1 - Ієрархічна структура вхідних даних

Рівень	Складові	Призначення
1	Дані датчика DHT11	Температура та відносна вологість
2	Дані датчика MQ135	Показник якості повітря
3	Службові дані	Wi-Fi, Blynk Token, параметри сеансу

Таблиця 2.2 - Ієрархічна структура вихідних даних

Рівень	Складові	Призначення
1	LCD 16×2	Локальне відображення параметрів
2	Blynk V0/V1/V2	Передавання даних у хмарний сервіс
3	pollution_alert	Аварійне сповіщення користувача

Віддалене відображення реалізовано через платформу Blynk IoT. Для передачі даних використовуються віртуальні канали V0, V1 та V2. Канал V0 використовується для передачі температури, канал V1 — для передачі відносної вологості, а канал V2 — для передачі показників якості повітря. Завдяки використанню віртуальних каналів забезпечується структуроване представлення інформації на вебпанелі та в мобільному застосунку.

Метод організації обміну даними побудований на циклічному принципі. Після завершення початкової ініціалізації система переходить у режим безперервного опитування датчиків. Передавання даних до хмарного сервісу виконується кожні 30 секунд за допомогою таймера BlynkTimer. Такий інтервал забезпечує достатню актуальність інформації та водночас зменшує навантаження на мережу передачі даних.

2.3 Розробка алгоритму

Одним із найважливіших етапів створення програмного забезпечення є розробка алгоритму його функціонування. Алгоритм визначає послідовність виконання операцій, порядок взаємодії окремих програмних модулів та механізми обробки вхідних і вихідних даних. Для розробленої IoT-системи моніторингу повітря алгоритм забезпечує безперервний контроль параметрів навколишнього середовища, передачу даних до хмарного сервісу Blynk IoT та інформування користувача про стан повітряного середовища.

2.3.1 Зовнішнє проектування програми

Зовнішнє проектування програми визначає набір функцій, доступних з точки зору користувача та апаратного оточення. Система працює як автономний вимірювально-інформаційний комплекс, що збирає дані з датчиків, відображає їх локально на дисплеї та передає до Blynk IoT.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						26
Змн.	Арк	№ докум.	Підпис	Дата		

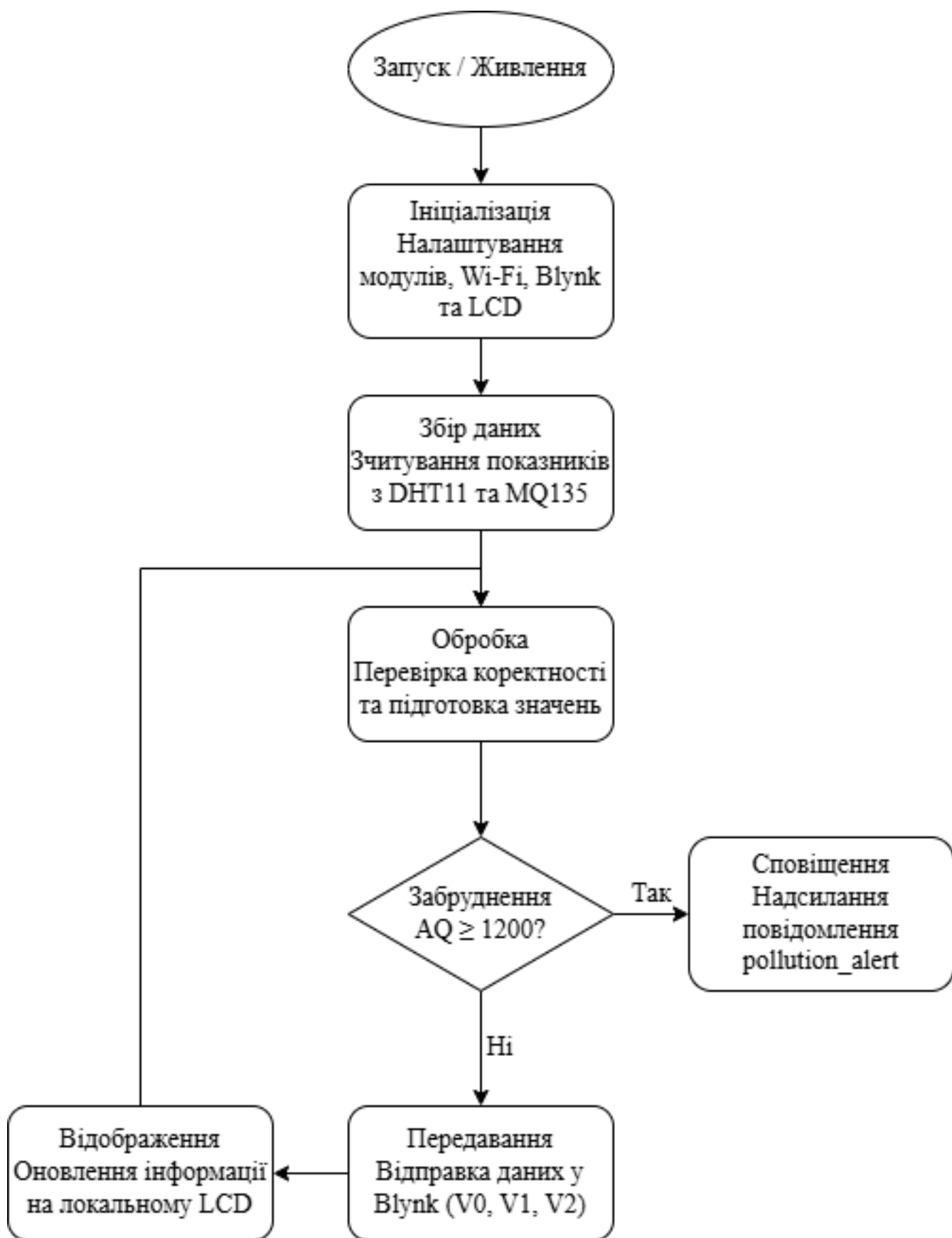


Рисунок 2.1 – Ієрархічна схема зовнішніх функцій програмного забезпечення

Таблиця 2.3 - Функціональні специфікації верхнього рівня

Функція	Вхід	Вихід	Реакція системи
Ініціалізація	Живлення	Готовність до роботи	Налаштування модулів, Wi-Fi, Blynk і дисплея
Збір даних	Сигнали DHT11, MQ135	Сирі вимірювання	Зчитування поточних показників
Обробка	Сирі вимірювання	Підготовлені значення	Перевірка коректності та порівняння
Відображення	Підготовлені значення	Інформація на LCD	Оновлення локального дисплея
Передавання	Підготовлені значення	Дані в Blynk	Оновлення V0, V1, V2
Сповіщення	$AQ \geq 1200$	pollution_alert	Надсилання повідомлення користувачу

Функції програми реалізовані за принципом зверху вниз. На верхньому рівні виділено групи функцій збору даних, обробки, відображення та сповіщення. На нижчому рівні ці групи деталізуються до окремих дій: читання сенсорів, перевірка коректності, передавання даних, формування події та виведення текстових повідомлень.

З точки зору користувача програма виконує такі дії: після запуску автоматично підключається до мережі, відображає параметри повітря на дисплеї, передає їх у мобільний застосунок та інформує про забруднення повітря. Поведінка програми залишається зрозумілою та передбачуваною за будь-яких вхідних даних, включно з некоректними показниками датчика.

Зовнішній рівень функціонування програми відповідає вимогам до IoT-систем реального часу, оскільки не потребує втручання користувача для отримання даних і забезпечує їх регулярне оновлення.

2.3.2 Проектування логіки програми

Логіка роботи програмного забезпечення реалізована у вигляді циклічного алгоритму, який забезпечує безперервний збір, обробку та передачу даних. Основу програмної реалізації складають функції `setup()` (див. рисунок 2.2), `loop()` та `sendSensor()`.

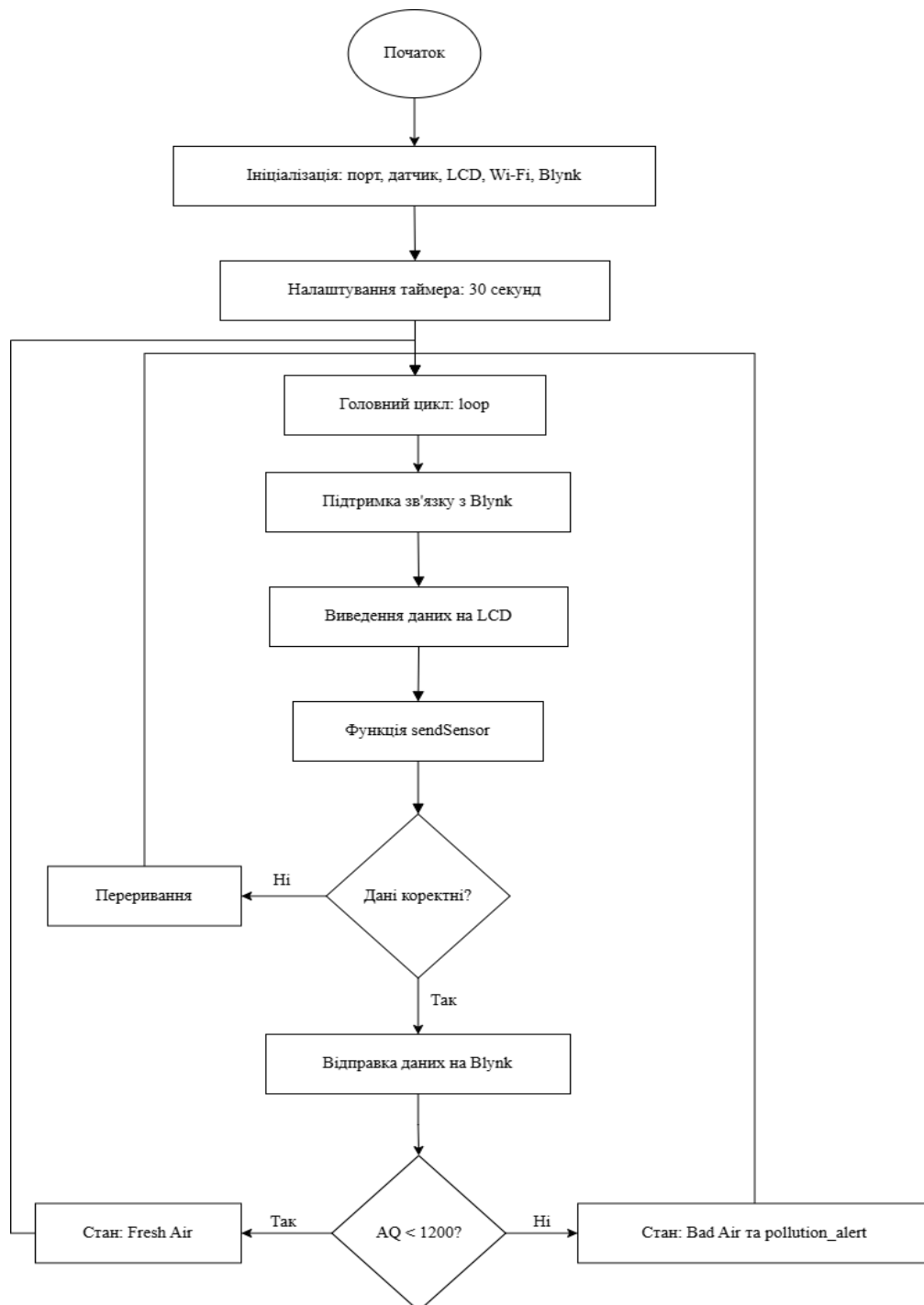


Рисунок 2.2 – Блок-схема функції `setup()`

Змн.	Арк	№ докум.	Підпис	Дата

Після подачі живлення виконується функція `setup()`, у якій здійснюється початкова конфігурація системи. На цьому етапі налаштовується послідовний інтерфейс обміну даними, виконується підключення до мережі Wi-Fi та сервера Blynk IoT, ініціалізується датчик DHT11 і LCD-дисплей. Після завершення ініціалізації на дисплеї виводиться повідомлення про запуск системи моніторингу. Далі програма переходить до виконання функції `loop()`, яка працює безперервно протягом усього часу функціонування пристрою.

У межах цього циклу забезпечується підтримка з'єднання з сервером Blynk та виконання програмного таймера. Зчитування даних із датчиків виконується функцією `sendSensor()`, яка автоматично запускається кожні 30 секунд. У процесі її виконання отримуються значення температури, вологості та рівня забруднення повітря. Перед подальшою обробкою здійснюється перевірка коректності даних датчика DHT11. Якщо отримані значення є помилковими, цикл вимірювання завершується без передачі інформації.

Після успішного зчитування даних виконується передача показників до платформи Blynk IoT через віртуальні канали:

V0 — температура повітря;

V1 — відносна вологість;

V2 — показник якості повітря.

Одночасно з передачею інформації здійснюється виведення результатів на LCD-дисплей. Для економії обмеженого простору дисплея використовується почергове відображення параметрів. Спочатку користувачу демонструється температура повітря, потім відносна вологість, після чого відображається поточний стан якості повітря.

Для оцінки якості повітря використовується пороговий метод. У програмі встановлено контрольне значення, яке дорівнює 1200 одиницям аналого-цифрового перетворення. Якщо показник MQ135 не перевищує встановлений поріг, система визначає стан повітря як нормальний та виводить повідомлення «Fresh Air».

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						30
Змн.	Арк	№ докум.	Підпис	Дата		

У випадку перевищення порогового значення система визначає повітря як забруднене та виводить повідомлення «Bad Air».

Під час виявлення забрудненого повітря виконується формування події `pollution_alert`. Створена подія автоматично надсилається до сервера Blynk IoT, після чого користувач отримує push-повідомлення через мобільний застосунок. Це забезпечує своєчасне інформування про погіршення стану повітряного середовища.



Рисунок 2.3 – Вікно програми Air Quality Monitoring

На рисунку 2.3 зображено вікно програми Air Quality Monitoring. Логічну структуру програмного забезпечення можна подати у вигляді таких функціональних модулів:

- модуль підключення до Wi-Fi;
- модуль взаємодії з Blynk IoT;
- модуль зчитування даних DHT11;
- модуль зчитування даних MQ135;
- модуль відображення інформації на LCD-дисплеї;
- модуль аналізу якості повітря;
- модуль формування повідомлень та сповіщень.

2.4 Визначення інформаційних зв'язків

Під час розробки програмного забезпечення IoT-системи моніторингу повітря було визначено інформаційні зв'язки між усіма компонентами системи, які забезпечують збір, обробку, передавання та відображення даних.

Таблиця 2.4 - Інформаційні зв'язки програмних компонентів

Джерело	Приймач	Дані	Тип зв'язку	Період
DHT11	ESP32	Температура , вологість	Інформаційн ий	30 с
MQ135	ESP32	ADC-рівень якості повітря	Інформаційн ий	30 с
ESP32	LCD 16×2	Строки відображенн я	Керувальний	постій но
ESP32	Blynk V0/V1/V2	Виміряні параметри	Інформаційн ий	30 с
ESP32	Blynk event	pollution_aler t	Подієвий	за порого м
Blynk	Мобільний застосунок	Графіки, індикатори, сповіщення	Зворотний інформаційн ий	у реальн ому часі

Центральним елементом системи є мікроконтролер XIAO ESP32C6, який виконує функції збору, обробки та передачі інформації. Перший інформаційний зв'язок реалізовано між датчиком DHT11 та мікроконтролером ESP32.

Датчик вологості повітря і передає отримані значення через цифровий інтерфейс. Програмне забезпечення виконує зчитування цих даних та їх подальшу обробку. Отримана інформація використовується для відображення на дисплеї та передачі до хмарного сервісу.

Другий інформаційний зв'язок реалізовано між датчиком MQ135 та ESP32. Датчик формує аналоговий сигнал, який залежить від рівня забруднення повітря. За допомогою вбудованого аналого-цифрового перетворювача мікроконтролер виконує перетворення аналогового сигналу в цифрове значення, яке надалі використовується для оцінювання якості повітряного середовища.

Після отримання даних від сенсорів формується внутрішній інформаційний потік між модулем збору даних та модулем їх обробки. На цьому етапі здійснюється перевірка коректності вимірювань, аналіз отриманих значень та визначення поточного стану повітряного середовища. Результатом обробки є набір підготовлених даних, придатних для подальшого використання.

Наступний інформаційний зв'язок реалізовано між ESP32 та LCD-дисплеєм. Передача даних здійснюється через інтерфейс I2C. На дисплей виводяться поточні значення температури, вологості та інформація про якість повітря. Такий зв'язок забезпечує локальне відображення результатів вимірювання без необхідності використання додаткових пристроїв.

Важливим елементом системи є зв'язок між мікроконтролером та платформою Blynk IoT. Передавання даних здійснюється через мережу Wi-Fi з використанням протоколів обміну, підтримуваних платформою Blynk. Для кожного параметра використовується окремий віртуальний канал:

V0 – температура повітря;

V1 – відносна вологість повітря;

V2 – показник якості повітря.

Через зазначені канали система регулярно передає результати вимірювань до хмарного сервісу, де вони використовуються для формування графіків, індикаторів та журналів спостережень.

Окремий інформаційний зв'язок реалізовано між платформою Blynk IoT та

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						33
Змн.	Арк	№ докум.	Підпис	Дата		

мобільним застосунком користувача. Дані, отримані від ESP32, автоматично відображаються на панелі моніторингу в режимі реального часу. Це дозволяє користувачеві контролювати стан повітряного середовища незалежно від місця знаходження пристрою.

Крім передачі вимірювальних даних, реалізовано канал передачі службових повідомлень. Якщо значення датчика MQ135 перевищує встановлене порогове значення, програмне забезпечення формує подію `pollution_alert`, яка передається до сервера Blynk. Після обробки події користувач отримує `push`-повідомлення про погіршення якості повітря. Таким чином забезпечується оперативне реагування на виникнення небезпечних умов.

Визначені інформаційні зв'язки забезпечують повний цикл функціонування системи: від збору інформації про стан повітряного середовища до її відображення, зберігання та передачі користувачеві. Така організація інформаційного обміну забезпечує надійність роботи системи, своєчасне оновлення даних та можливість віддаленого моніторингу параметрів мікроклімату приміщення.

2.5 Написання текстів програм

Після завершення етапів проектування було виконано написання текстів програм мовою C++ у середовищі Arduino IDE. Для реалізації системи використано бібліотеки `'WiFi.h'`, `'BlynkSimpleEsp32.h'`, `'DHT.h'` та `'LiquidCrystal_I2C.h'`, які забезпечують підключення до мережі, взаємодію з хмарною платформою, роботу з датчиком DHT11 і керування дисплеєм I2C.

Структура програмного коду відповідає модульному принципу. Окремі фрагменти коду відповідають за ініціалізацію системи, зчитування показників датчиків, обробку значень, локальне відображення, передавання даних у Blynk та формування події про забруднення повітря. Така організація робить програму зрозумілою та придатною до супроводу.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						34
Змн.	Арк	№ докум.	Підпис	Дата		

Нижче представлено програмний код підключення бібліотек та оголошення змінних

```
#define BLYNK_TEMPLATE_ID "Your_Temp_ID"
#define BLYNK_TEMPLATE_NAME "Air Quality Monitoring"
#define BLYNK_AUTH_TOKEN "Your_Auth_Token"

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <DHT.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);
#define gas A0
#define DHTPIN D2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
```

Нижче представлено програмний код функція зчитування та передавання даних

```
void sendSensor()
{
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    int analogSensor = analogRead(gas);

    if (isnan(h) || isnan(t)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    Blynk.virtualWrite(V0, t);
    Blynk.virtualWrite(V1, h);
    Blynk.virtualWrite(V2, analogSensor);
```



					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						35
Змн.	Арк	№ докум.	Підпис	Дата		

У програмі реалізовано перевірку коректності значень датчика DHT11, а також порогову логіку для MQ135. Результати вимірювань передаються на віртуальні канали V0, V1 та V2, що дозволяє відображати їх у вебпанелі та мобільному застосунку Blynk IoT. При перевищенні порога формується подія `'pollution_alert'`, яка використовується для надсилання повідомлення користувачеві.

Написаний програмний текст відповідає вимогам до вбудованих систем: він є відносно компактним, використовує стандартні бібліотеки, має зрозумілу структуру та забезпечує незалежну роботу основних функціональних модулів.

2.6 Тестування та налагодження програм

Після завершення розробки програмного забезпечення було проведено тестування та налагодження IoT-системи моніторингу повітря на базі мікроконтролера XIAO ESP32C6. Метою тестування була перевірка працездатності всіх програмних модулів, правильності обробки даних від датчиків, коректності передачі інформації до хмарної платформи Blynk IoT та стабільності роботи системи в цілому.

Налагодження програмного забезпечення виконувалося поетапно. На першому етапі здійснювалася перевірка правильності компіляції програмного коду та усунення синтаксичних помилок. Для цього використовувалося середовище розробки Arduino IDE, яке дозволило контролювати процес компіляції та завантаження програми до мікроконтролера.

Після успішного завантаження програми було проведено тестування підключення до бездротової мережі Wi-Fi. У процесі перевірки контролювалося встановлення з'єднання з точкою доступу та коректна авторизація на сервері Blynk IoT. Результати тестування підтвердили стабільне підключення пристрою до мережі та успішний обмін даними з хмарною платформою.

На рисунку 2.5 зображено вікно відображення результатів

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						36
Змн.	Арк	№ докум.	Підпис	Дата		



Рисунок 2.5 – Вікно відображення результатів

Наступним етапом стала перевірка роботи датчика температури та вологості DHT11. Для цього виконувалося порівняння значень, отриманих від датчика, з фактичними параметрами навколишнього середовища. Отримані результати підтвердили коректність зчитування даних та їх правильне відображення на LCD-дисплеї й у мобільному застосунку Blynk.

Окрему увагу було приділено перевірці роботи датчика якості повітря MQ135. Під час тестування здійснювалося спостереження за зміною показників датчика при зміні складу повітряного середовища. Отримані значення коректно змінювалися відповідно до умов експерименту та передавалися до системи моніторингу без помітних затримок.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						37
Змн.	Арк	№ докум.	Підпис	Дата		

Для перевірки працездатності функції відображення інформації було виконано тестування LCD-дисплея. У ході перевірки контролювалося коректне виведення температури, вологості та повідомлень про якість повітря. Встановлено, що інформація відображається без спотворень та своєчасно оновлюється відповідно до нових результатів вимірювання. На рисунку 2.6 зображено тетування LCD-дисплею.



Рисунок 2.6 – Тетування LCD-дисплею.

Під час тестування системи також перевірялася передача даних до платформи Blynk IoT. Контроль здійснювався через вебпанель та мобільний застосунок. Результати показали, що значення температури, вологості та якості повітря успішно надходять до відповідних віртуальних каналів та відображаються в режимі реального часу.

Для перевірки механізму сповіщень було виконано тестування функції формування події pollution_alert. Під час штучного перевищення встановленого порогового значення датчика MQ135 система успішно формувала подію та передавала її на сервер Blynk IoT. У результаті користувач отримував push-повідомлення про погіршення якості повітря.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						38
Змн.	Арк	№ докум.	Підпис	Дата		

Результати проведеного тестування наведено в таблиці 2.5.

Об'єкт тестування	Тестові умови	Очікуваний результат	Фактичний результат
Підключення до Wi-Fi	Ввімкнення пристрою	Успішна авторизація	Виконано успішно
Підключення до Blynk IoT	Наявність інтернету	Синхронізація з сервером	Виконано успішно
Датчик DHT11	Нормальні умови в приміщенні	Отримання температури та вологості	Виконано успішно
Датчик MQ135	Зміна якості повітря	Зміна показника AQ	Виконано успішно
LCD-дисплей	Передача тестових даних	Коректне відображення інформації	Виконано успішно
Повідомлення pollution_alert	$AQ \geq 1200$	Push-сповіщення користувачу	Виконано успішно

Таблиця 2.5 – Результати тестування програмного забезпечення

Для оцінювання стабільності роботи система була переведена в режим тривалого функціонування. Протягом декількох годин безперервної роботи не було виявлено збоїв у роботі датчиків, втрати мережевого з'єднання або помилок передавання даних. Усі модулі програмного забезпечення працювали відповідно до заданого алгоритму.

У процесі налагодження були усунуті окремі недоліки, пов'язані з відображенням інформації на LCD-дисплеї та початковим підключенням до мережі Wi-Fi. Після внесення необхідних коригувань система почала функціонувати стабільно та без помилок.

Таким чином, проведене тестування підтвердило працездатність розробленого програмного забезпечення та його відповідність поставленим вимогам. Результати перевірки засвідчили коректну роботу всіх функціональних модулів системи моніторингу повітря, стабільність передачі даних до платформи Blynk IoT та надійність функціонування системи в режимі реального часу.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						39
Змн.	Арк	№ докум.	Підпис	Дата		

3 СПЕЦІАЛЬНИЙ РОЗДІЛ

3.1 Інструкція з інсталяції та налаштування вебсерверів та супутнього програмного забезпечення

Для забезпечення функціонування розробленого програмного комплексу IoT-системи моніторингу повітря використовується хмарна платформа Blynk IoT, середовище програмування Arduino IDE та програмне забезпечення для підключення мікроконтролера XIAO ESP32C6 до бездротової мережі Wi-Fi. Оскільки система реалізує передачу даних через Інтернет і локальне відображення інформації на LCD-дисплеї, окремий локальний вебсервер на стороні користувача не потрібний, а функції віддаленого доступу виконує хмарний сервіс Blynk.

Мінімальна конфігурація технічних засобів для інсталяції та запуску програмного комплексу включає персональний комп'ютер або ноутбук з операційною системою Windows 10/11, Linux або macOS, мікроконтролерну плату XIAO ESP32C6, датчик температури та вологості DHT11, датчик якості повітря MQ135, LCD-дисплей 16×2 з I2C-перетворювачем, макетну плату, з'єднувальні проводи та кабель для підключення мікроконтролера до комп'ютера. Для стабільної роботи необхідно також мати доступ до мережі Wi-Fi та обліковий запис у сервісі Blynk IoT. Середовище програмування, використане під час розробки програмного забезпечення, — Arduino IDE. Воно забезпечує створення, компіляцію та завантаження програмного коду до плати XIAO ESP32C6. Для коректної роботи з даною платформою в середовище попередньо встановлюється підтримка плат ESP32 через менеджер плат Arduino. Крім того, до проєкту підключаються бібліотеки WiFi.h, BlynkSimpleEsp32.h, DHT.h та

LiquidCrystal_I2C.h, які забезпечують реалізацію мережевої взаємодії, обміну даними з Blynk, роботи з датчиком DHT11 та керування LCD-дисплеєм відповідно.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						40
Змн.	Арк	№ докум.	Підпис	Дата		

Інсталяція супутнього програмного забезпечення виконується послідовно. Спочатку встановлюється Arduino IDE, після чого у параметрах програми додається посилання на менеджер плат ESP32. Далі через менеджер бібліотек інсталиються необхідні бібліотеки для роботи з апаратними компонентами системи. Після цього виконується налаштування середовища Blynk IoT через вебінтерфейс сервісу. У хмарній платформі створюється шаблон пристрою, визначаються віртуальні канали для передавання температури, вологості та показників якості повітря, а також налаштовуються події та сповіщення.

Для роботи програмного комплексу в Blynk IoT створюються такі налаштувальні файли та параметри:

- шаблон пристрою з назвою Air Quality Monitoring;
- віртуальні канали V0, V1, V2;
- подія pollution_alert;
- токен автентифікації пристрою;
- параметри доступу до Wi-Fi-мережі.

У програмному коді ці параметри задаються у вигляді констант та змінних конфігурації. Зокрема, передбачено введення ідентифікатора шаблону, назви шаблону, токена доступу, імені мережі Wi-Fi та пароля. Таке рішення забезпечує можливість швидкого повторного налаштування пристрою без зміни основної логіки програми.

Мова програмування комплексу — C++ із використанням бібліотек платформи Arduino. Технологічний тип представлення програми — початковий текстовий код, що зберігається у файлі формату .ino. Після компіляції формується завантажувальний модуль, який передається до мікроконтролера XIAO ESP32C6. За необхідності у процесі розробки можуть також використовуватися об'єктні та проміжні файли, що створюються середовищем компіляції автоматично.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						41
Змн.	Арк	№ докум.	Підпис	Дата		

Після завершення інсталяції та налаштування програмного забезпечення виконується перевірка працездатності системи. Для цього мікроконтролер підключається до комп'ютера, програма завантажується до плати, після чого перевіряється з'єднання з мережею Wi-Fi, робота датчиків, відображення даних на LCD-дисплеї та передавання інформації до Blynk IoT. У разі успішного виконання всіх етапів інсталяції програмний комплекс вважається готовим до експлуатації.

Таким чином, інсталяція та налаштування вебсервісу Blynk IoT і супутнього програмного забезпечення забезпечують повну підготовку IoT-системи моніторингу повітря до роботи в реальних умовах. Використання Arduino IDE, бібліотек для ESP32 та хмарної платформи Blynk дає змогу реалізувати стабільний обмін даними, локальне відображення показників і віддалений контроль стану повітряного середовища.

3.2 Інструкція по встановленню та налаштуванню розробленого проєкту

Встановлення та налаштування розробленого проєкту виконується поетапно і включає монтаж апаратної частини, підключення елементів схеми, завантаження програмного коду до мікроконтролера, а також налаштування хмарної платформи Blynk IoT для віддаленого моніторингу параметрів повітря. Розроблений програмно-апаратний комплекс призначений для роботи на базі плати XIAO ESP32C6 та забезпечує вимірювання температури, вологості й рівня забруднення повітря з подальшим відображенням даних на LCD-дисплеї та передаванням їх до мобільного застосунку.

Першим етапом встановлення є підготовка всіх необхідних компонентів системи. Для складання проєкту використовуються: плата XIAO ESP32C6, датчик температури та вологості DHT11, датчик якості повітря MQ135, LCD- дисплей 16×2 з I2C-модулем, макетна плата та з'єднувальні проводи. Перед початком монтажу необхідно переконатися у справності всіх компонентів і наявності джерела живлення та кабелю для підключення мікроконтролера до комп'ютера.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						42
Змн.	Арк	№ докум.	Підпис	Дата		

Монтаж апаратної частини виконується на макетній платі. Датчик DHT11 підключається до цифрового входу, визначеного в програмному коді як D2. Вивід живлення датчика з'єднується з лінією живлення 3,3 В або 5 В залежно від модуля, а спільний контакт — із загальною шиною GND. Датчик MQ135 підключається до аналогового входу A0, що дозволяє зчитувати значення рівня забруднення повітря. LCD-дисплей з I2C-перетворювачем підключається до відповідних ліній інтерфейсу I2C: SDA, SCL, а також до живлення та землі.

Після фізичного з'єднання всіх елементів необхідно перевірити правильність підключення проводів відповідно до схеми. Особливу увагу слід приділити полярності живлення, надійності контактів на макетній платі та правильності підключення сигнальних ліній. Неправильне з'єднання може призвести до некоректної роботи датчиків або пошкодження елементів схеми.

Наступним етапом є підготовка програмного середовища. На комп'ютері встановлюється Arduino IDE, після чого додається підтримка плат ESP32. У середовищі розробки відкривається програмний файл проєкту, у якому вказуються назва мережі Wi-Fi, пароль доступу, ідентифікатор шаблону Blynk, назва шаблону та токен автентифікації пристрою. Після цього перевіряється правильність підключення необхідних бібліотек і виконується компіляція проєкту.

Далі здійснюється налаштування хмарної платформи Blynk IoT. У вебінтерфейсі створюється новий шаблон пристрою з назвою Air Quality Monitoring. Для цього вказується тип обладнання ESP32 та тип з'єднання Wi-Fi. Після створення шаблону додаються три канали даних:

- V0 — температура;
- V1 — вологість;
- V2 — якість повітря.

Для кожного каналу задається тип даних і граничні значення, що використовуються у віджетах мобільного застосунку та вебпанелі. Після цього створюється подія `pollution_alert`, яка використовується для надсилання повідомлень користувачу у разі перевищення допустимого рівня забруднення повітря.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						43
Змн.	Арк	№ докум.	Підпис	Дата		

Після створення шаблону та каналів даних необхідно сформувати пристрій у Blynk IoT на основі створеного шаблону. Під час створення пристрою система генерує токен автентифікації, який копіюється до програмного коду. Саме цей токен забезпечує підключення плати XIAO ESP32C6 до хмарної платформи та передачу даних у режимі реального часу.

Після налаштування Blynk виконується завантаження програми до мікроконтролера. Для цього в Arduino IDE вибирається відповідна плата XIAO ESP32C6 та потрібний COM-порт. Після запуску компіляції програмний код передається до пристрою. У разі успішного завантаження плата автоматично підключається до заданої Wi-Fi-мережі та починає роботу згідно з алгоритмом програми.

Після запуску системи на LCD-дисплеї з'являється стартове повідомлення про початок роботи моніторингу. Далі на екран по черзі виводяться значення температури, вологості та стану якості повітря. Одночасно ці ж дані передаються на сервер Blynk IoT, де відображаються у відповідних віджетах мобільного застосунку або вебпанелі.

Для перевірки працездатності після встановлення виконується контрольний запуск системи. На цьому етапі перевіряється:

- наявність з'єднання з мережею Wi-Fi;
- коректність зчитування даних з датчика DHT11;
- коректність зчитування даних з датчика MQ135;
- відображення інформації на LCD-дисплеї;
- передавання значень до Blynk IoT;
- формування сповіщення `pollution_alert` при перевищенні порогу забруднення.

Якщо всі зазначені функції працюють коректно, проєкт вважається встановленим і налаштованим. У разі виявлення помилок виконується повторна перевірка підключення датчиків, правильності конфігураційних параметрів Wi-Fi та Blynk, а також відповідності апаратних виводів значенням, зазначеним у програмі.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						44
Змн.	Арк	№ докум.	Підпис	Дата		

Таким чином, процес встановлення та налаштування розробленого проекту забезпечує повну готовність IoT-системи до експлуатації. Після завершення всіх етапів користувач отримує працездатний програмно-апаратний комплекс, який здійснює безперервний моніторинг повітря, локальне відображення показників та дистанційне інформування про стан повітряного середовища

3.3 Інструкція з використання тестових наборів

Для перевірки працездатності розробленого програмного комплексу IoT-системи моніторингу повітря було сформовано набір тестів, який охоплює основні режими роботи системи, граничні стани, нештатні ситуації та взаємодію між окремими програмними й апаратними компонентами. Використання тестових наборів дає змогу перевірити коректність зчитування даних з датчиків DHT11 і MQ135, правильність відображення інформації на LCD-дисплеї, стабільність передавання даних до Blynk IoT, а також роботу механізму сповіщення про перевищення допустимого рівня забруднення повітря.

Під час тестування було враховано вимоги методичних вказівок щодо застосування таких типів тестів, як аварійні, вироджені, комплексні, основні, стикові, структурні та функціональні. Кожен із зазначених тестів використовується для перевірки окремого аспекту роботи системи та дозволяє виявити можливі помилки у різних частинах програмного забезпечення. Основні тести застосовуються для перевірки звичайного режиму роботи системи. У цих умовах усі апаратні модулі працюють штатно, а програмне забезпечення повинно забезпечувати безперервне зчитування показників, їх відображення та передавання до хмарного сервісу. Саме ці тести підтверджують, що розроблений комплекс відповідає поставленій задачі та виконує свої функції у стандартних умовах.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						45
Змн.	Арк	№ докум.	Підпис	Дата		

Основні тестові набори для розробленого проєкту подано в таблиці 3.1.

Таблиця 3.1 – Тестові набори для перевірки програмного комплексу

№	Тип тесту	Умови виконання	Очікуваний результат
1	Основний тест	Нормальна робота систем коректне підключення датчиків, стабільне живлення та наявність Wi- в	и, Отримання значень температури, вологості та якості повітря, відображення їх на LCD і Fi в Blynk
2	Функціональний тест	Перевірка окремої функції зчитування DHT11	Коректне відображення температури та вологості
3	Функціональний тест	Перевірка окремої функції зчитування MQ135	Коректне отримання значення якості повітря
4	Структурний тест	Перевірка зв'язків між функціями setup(), loop() та sendSensor()	Коректне виконання циклу ініціалізації, збору та передавання даних
5	Стиковий тест	Взаємодія датчиків, LCD-дисплея та Blynk IoT	Коректний обмін даними між усіма модулями
6	Комплексний тест	Повний цикл роботи системи в реальних умовах	Стабільне функціонування всіх підсистем без збоїв
7	Аварійний тест	Відключення датчика DHT11 або порушення його роботи	Виявлення помилки та відсутність передавання некоректних даних
8	Аварійний тест	Відсутність з'єднання з Wi-Fi	Пристрій не завершує роботу аварійно, а продовжує локальну роботу
9	Вироджений тест	Значення MQ135 на межі порогу спрацювання	Коректне визначення стану повітря без помилкового перемикавання
10	Вироджений тест	Значення температури або вологості близькі до граничних значень датчика	Коректна інтерпретація отриманих даних або фіксація помилки вимірювання

Функціональні тести призначені для перевірки окремих програмних функцій незалежно одна від одної. У даному проєкті окремо перевіряється робота підпрограми зчитування температури та вологості, функції обробки сигналу від MQ135, а також функції передавання даних на віртуальні канали Blynk. Такі тести

дозволяють виявити помилки в окремих модулях без необхідності запускати всю систему повністю.

Структурні тести використовуються для перевірки правильності внутрішньої організації програмного коду та взаємодії між його частинами. Для розробленої системи це означає контроль логічного зв'язку між функціями ініціалізації, збору даних, обробки інформації та відображення результатів. Особлива увага приділяється коректності переходу від одного етапу роботи до іншого.

Стикові тести застосовуються для перевірки сумісності програмних і апаратних модулів. У даному випадку перевіряється правильність взаємодії між платою XIAO ESP32C6, датчиками DHT11 і MQ135, LCD-дисплеєм та сервісом Blynk IoT. Такі тести дають змогу переконатися, що всі компоненти системи працюють як єдине ціле.

Комплексні тести виконуються в умовах, максимально наближених до реальної експлуатації. Під час таких випробувань система працює протягом тривалого часу, а контроль здійснюється за всіма параметрами одночасно. Це дозволяє оцінити не лише коректність роботи окремих елементів, а й загальну стабільність програмного комплексу.

Аварійні тести проводяться для перевірки поведінки системи у випадку виникнення нештатних ситуацій. До таких ситуацій належать обрив лінії з датчиком, відсутність з'єднання з мережею Wi-Fi, помилки під час обміну даними або вихід показників за допустимі межі. У ході аварійного тестування система повинна зберігати базову працездатність і, за можливості, повідомляти користувача про проблему.

Вироджені тести застосовуються для перевірки граничних значень вхідних даних. Для даного проєкту це значення, що знаходяться поблизу порогу визначення якості повітря, а також мінімальні та максимальні показники, які може видати датчик DHT11 або MQ135. Метою таких тестів є перевірка того, що програма правильно обробляє критичні значення та не допускає помилкової інтерпретації даних.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						47
Змн.	Арк	№ докум.	Підпис	Дата		

Під час виконання тестів використовувалося таке правило: кожен тестовий набір запускався окремо, результати фіксувалися у процесі роботи системи, після чого порівнювалися з очікуваними значеннями. У разі невідповідності виконувалося уточнення причин помилки та повторна перевірка відповідного модуля.

Отже, використання тестових наборів забезпечило всебічну перевірку розробленого програмного комплексу. Застосовані типи тестів дали змогу підтвердити правильність функціонування системи моніторингу повітря, стабільність передавання даних, коректність обробки сигналів від датчиків та надійність інформування користувача про стан повітряного середовища.

Якщо потрібно, я одразу зроблю 3.4 Інструкція з експлуатації програмного комплексу у такому ж стилі, щоб спеціальний розділ був завершений повністю.

3.4 Інструкція з експлуатації програмного комплексу

Програмний комплекс IoT-системи моніторингу повітря призначений для безперервної роботи в автоматичному режимі після його інсталяції, налаштування та завантаження до мікроконтролера XIAO ESP32C6. Експлуатація системи не потребує спеціальної підготовки користувача, оскільки основні функції моніторингу, відображення показників та передавання даних до хмарної платформи Blynk IoT виконуються автоматично.

Перед початком експлуатації необхідно переконатися, що апаратна частина зібрана відповідно до схеми підключення, датчики DHT11 і MQ135 підключені до відповідних виводів мікроконтролера, LCD-дисплей із I2C-модулем правильно з'єднаний із платою, а пристрій має стабільне живлення. Також слід перевірити наявність доступу до мережі Wi-Fi та коректність внесених параметрів підключення у програмний код, зокрема імені мережі, пароля, ідентифікатора шаблону Blynk, назви шаблону та токена автентифікації.

Запуск програмного комплексу здійснюється шляхом подачі живлення на плату XIAO ESP32C6. Після увімкнення пристрій автоматично виконує

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						48
Змн.	Арк	№ докум.	Підпис	Дата		

ініціалізацію датчиків, встановлює з'єднання з Wi-Fi, підключається до сервера Blynk IoT та переходить у робочий режим. На LCD-дисплеї після запуску відображається службове повідомлення про початок роботи системи моніторингу. У подальшому на дисплей по чергово виводяться поточні значення температури, відносної вологості та стану якості повітря.

Під час роботи системи дані з датчиків періодично оновлюються та передаються до хмарної платформи Blynk IoT. У мобільному застосунку або на вебпанелі користувач має можливість спостерігати значення температури, вологості та рівня забруднення повітря в режимі реального часу. Якщо значення, отримане з датчика MQ135, перевищує встановлений поріг, система автоматично формує подію `pollution_alert`, після чого користувач отримує сповіщення про погіршення якості повітря.

У процесі експлуатації не потрібно вручну запускати окремі програмні модулі, оскільки весь комплекс працює як єдина система. Користувач може лише контролювати інформацію, що надходить на дисплей і до Blynk, а також за потреби змінювати параметри доступу до мережі або конфігураційні значення в програмному коді перед повторним завантаженням до пристрою. У разі тимчасової втрати з'єднання з Wi-Fi пристрій продовжує локальну роботу, відображаючи показники на LCD-дисплеї, а після відновлення мережі автоматично поновлює передавання даних до сервісу Blynk.

Командні файли комплексу зберігаються у вигляді початкового текстового файлу проєкту формату `.ino`. Додаткові програмні компоненти, а саме бібліотеки для роботи з Wi-Fi, Blynk, DHT11 та LCD-дисплеєм, підключаються у вигляді стандартних бібліотек Arduino та розміщуються у відповідних системних каталогах середовища розробки. Інформаційними файлами системи є конфігураційні значення, задані безпосередньо в програмному коді: `BLYNK_TEMPLATE_ID`, `BLYNK_TEMPLATE_NAME`, `BLYNK_AUTH_TOKEN`, `ssid`, `pass`, а також налаштування віртуальних каналів `V0`, `V1`, `V2`.

Програмний комплекс не потребує значних обсягів оперативної пам'яті, оскільки реалізує порівняно прості функції збору даних, їх обробки та

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						49
Змн.	Арк	№ докум.	Підпис	Дата		

передавання до хмарного сервісу. Основне навантаження припадає на бібліотеки мережевої взаємодії та роботу з дисплеєм, однак обсяг використовуваних ресурсів залишається в межах можливостей плати XIAO ESP32C6. Завантажувальний модуль формується середовищем Arduino IDE після компіляції програми та записується у флеш-пам'ять мікроконтролера. Обсяг цього модуля є невеликим і цілком відповідає технічним можливостям обраної апаратної платформи.

Для коректної експлуатації системи рекомендується дотримуватись таких умов: не допускати механічного пошкодження датчиків, забезпечувати стабільне живлення, уникати роботи в умовах надмірної вологості без захисного корпусу та регулярно перевіряти працездатність Wi-Fi-з'єднання. Після кожного перезапуску пристрою доцільно контролювати правильність відображення даних на LCD-дисплеї та наявність оновлення інформації у Blynk IoT.

Отже, експлуатація програмного комплексу зводиться до запуску пристрою, контролю його підключення до мережі та подальшого спостереження за показниками мікроклімату. Система працює автономно, забезпечує локальне та віддалене відображення результатів вимірювань, а також своєчасно інформує користувача про погіршення якості повітря в приміщенні. Якщо потрібно, я одразу можу зібрати для тебе весь 3-й розділ у єдиному стилі, щоб він був уже готовий до вставки в кваліфікаційну роботу.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						50
Змн.	Арк	№ докум.	Підпис	Дата		

4 ЕКОНОМІЧНИЙ РОЗДІЛ

4.1. Визначення стадій технологічного процесу та загальної тривалості проведення НДР

У цьому підрозділі визначаються основні стадії виконання науково-дослідної роботи (НДР) на тему «Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32». Метою роботи є створення програмного забезпечення для збору даних із датчиків якості повітря, їх передачі через бездротову мережу, аналізу отриманої інформації та автоматичного керування системою вентиляції.

Для оцінки трудомісткості проєкту визначимо основні етапи розробки та витрати часу на їх виконання. До виконання робіт залучаються: керівник проєкту, Embedded-розробник, Backend-розробник та інженер з тестування (QA).

Таблиця 4.1 - Середній час виконання робіт по стадіях технологічного процесу

№ п/п	Назва операції (стадії)	Виконавець	Середній час виконання операції, год.
1	Аналіз предметної області та існуючих IoT-рішень	Керівник проєкту	7
		Embedded-розробник	6
2	Формування технічного завдання та проєктування архітектури системи	Керівник проєкту	9
		Backend-інженер	6
		Embedded-розробник	8
3	Розробка програмного забезпечення для ESP32	Embedded-розробник	40
4	Розробка серверної частини та API	Backend-розробник	30

5	Реалізація алгоритмів моніторингу та керування вентиляцією	Backend-інженер	10
		Embedded-розробник	20
6	Інтеграція апаратної та програмної частин	Embedded-розробник	12
		Backend-інженер	8
7	Тестування та налагодження системи	Backend-інженер	4
		QA Engineer	14
		Embedded-розробник	6
8	Підготовка документації	Керівник проєкту	6
		Embedded-розробник	4
9	Управління проєктом та контроль виконання	Керівник проєкту	14
Разом			186

Сумарний час виконання операцій технологічного процесу становить 186 годин.

4.2. Визначення витрат на оплату праці та відрахувань на соціальні заходи

У цьому підрозділі здійснюється розрахунок витрат на оплату праці спеціалістів, задіяних у виконанні науково-дослідної роботи «Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32», а також відрахувань на соціальні заходи

Розмір заробітної плати визначається кваліфікацією працівника, складністю виконуваних робіт та рівнем відповідальності.

Для розрахунків приймемо такі погодинні ставки:

Керівник проєкту – 420 грн/год; Embedded-розробник – 380 грн/год; Backend-розробник – 340 грн/год; QA Engineer – 220 грн/год.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						52
Змн.	Арк	№ докум.	Підпис	Дата		

Основна заробітна плата (Зосн) визначається за формулою:

$$З_{осн.} = Т_c * К_г \quad (4.1)$$

де: T_c – тарифна ставка, грн/год; K_g – кількість відпрацьованих годин.

Отже, основна заробітна плата для:

Керівника проєкту: $Зосн1 = 420 * 36 = 15\,120$ грн.

Embedded-розробник: $Зосн2 = 380 * 86 = 32\,680$ грн.

Backend-розробник: $Зосн3 = 340 * 50 = 17\,000$ грн.

QA Engineer: $Зосн4 = 220 * 14 = 3\,080$ грн.

Сумарна основна заробітна плата становить:

$Зосн = 15\,120 + 32\,680 + 17\,000 + 3\,080 = 67\,880$ грн.

Додаткова заробітна плата становить 15% від суми основної заробітної плати.

$$З_{дод.} = З_{осн.} \cdot К_{додл.} \quad (4.2)$$

де: $K_{додл.}$ – коефіцієнт додаткових виплат працівникам.

Отже, додаткова заробітна плата по категоріях працівників становить:

Керівника проєкту: $Здод1 = 15\,120 * 0,15 = 2\,268$ грн.

Embedded-розробник: $Здод2 = 32\,680 * 0,15 = 4\,902$ грн.

Backend-розробник: $Здод3 = 17\,000 * 0,15 = 2\,550$ грн.

QA Engineer: $Здод4 = 3\,080 * 0,15 = 462$ грн

Загальна додаткова заробітна плата становить:

$Здод = 2\,268 + 4\,902 + 2\,550 + 462 = 10\,182$ грн

Звідси загальні витрати на оплату праці ($V_{о.п.}$) визначаються за формулою:

$$V_{о.п.} = З_{осн.} + З_{дод.} \quad (4.3)$$

$V_{оп} = 67\,880 + 10\,182 = 78\,062$ грн

Єдиний соціальний внесок (ЄСВ) становить 22 % від фонду оплати праці:

$$V_{есв} = V_{оп} * 0,22 \quad (4.4)$$

$V_{есв} = 78\,062 * 0,22 = 17\,173,64$ грн.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						53
Змн.	Арк	№ докум.	Підпис	Дата		

Проведені розрахунки витрат на оплату праці наведено у зведеній таблиці 4.2.

Таблиця 4.2 – Зведені розрахунки витрат на оплату праці

№ п/п	Категорія працівників	Основна заробітна плата, грн.			Додаткова заробітна плата, грн.	ЄСВ, грн.	Всього витрат и на оплату праці, грн. 6 = 3+4+5
		Тарифна ставка, грн.	К-сть годин	Фактично нарах. зарплати, грн.			
		1	2	3	4	5	6
1	Керівник проєкту	420	36	15 120,00	2 268,00	---	---
2	Embedded- розробник	380	86	32 680,00	4 902,00	---	---
3	Backend- розробник	340	50	17 000,00	2 550,00	---	---
4	QA Engineer	220	14	3 080,00	462,00	---	---
Разом				67 880,00	10 182,00	17 173,64	95 35,64

4.3 Розрахунок витрат на електроенергію

Під час виконання НДР «Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32» використовувалося комп'ютерне обладнання для програмування мікроконтролера ESP32, розробки серверної частини, тестування системи та підготовки документації.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						54
Змн.	Арк	№ докум.	Підпис	Дата		

Витрати на електроенергію визначаються за формулою:

$$Z_e = W * T * S \quad (4.5)$$

де: W – необхідна потужність, кВт. Для розробки використовувався один потужний комп'ютер, середня споживана потужність якого разом з монітором становить 0,45 кВт;

T – загальна кількість годин роботи обладнання; S
– вартість 1 кВт-години електроенергії.

$$Z_e = 0,45 * 186 * 15,94 = 1\,185,94 \text{ грн.}$$

4.4 Розрахунок суми амортизаційних відрахувань «Твоя квартира тут»

Амортизація являє собою процес поступового перенесення вартості основних засобів на собівартість продукції або послуг у міру їх використання. У процесі виконання НДР «Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32» використовувалося комп'ютерне обладнання, яке належить до четвертої групи основних засобів.

Для розрахунку амортизаційних відрахувань використовується формула:

$$A = \frac{БВ * Н_A}{100\%} * T \quad (4.6)$$

Де: A – амортизаційні відрахування за звітний період, грн.;

$БВ$ – балансова вартість групи основних фондів, грн.;

$Н_A$ – норма амортизації;

T – тривалість робіт у годинах.

Оскільки для створення та тестування веб-додатку використовується один персональний комп'ютер вартістю 70 000 грн, амортизаційні відрахування розраховуються за цією сумою і становитимуть:

$$A = \frac{70\,000 * 0,04}{150} * 186 = 3\,472 \text{ грн.}$$

					2026.КВР.123.405.04.00.00 ПЗ	Арк.
						55
Змн.	Арк	№ докум.	Підпис	Дата		

4.5 Обчислення накладних витрат

Накладні витрати — це витрати, пов'язані з організацією, управлінням та забезпеченням процесу виконання науково-дослідної роботи. До них належать витрати на адміністрування, використання мережевого обладнання, доступ до мережі Інтернет, комунальні послуги, програмне забезпечення, технічну підтримку та інші загальновиробничі витрати.

У практиці економічних розрахунків для розробки програмного забезпечення накладні витрати зазвичай становлять від 20% до 60% від фонду оплати праці. Для даної НДР приймаємо коефіцієнт накладних витрат на рівні 35%. Накладні витрати визначаються за формулою:

$$H_B = B_{o.p.} * K_H \quad (4.7)$$

Де: H_B – сума накладних витрат, грн.; $B_{o.p.}$ – загальні витрати на оплату праці. K_H – коефіцієнт накладних витрат.

$$H_B = 78\,062. * 0,35 = 27\,321,70 \quad \text{грн.}$$

4.6 Складання кошторису витрат та визначення собівартості НДР

Для визначення повної собівартості науково-дослідної роботи на тему «Розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32» необхідно звести всі попередньо розраховані витрати у єдиний кошторис. Це дозволяє оцінити загальну вартість розробки програмного продукту та визначити його економічну доцільність. На основі виконаних розрахунків сформовано кошторис витрат на виконання науково-дослідної роботи, який включає витрати на оплату праці, відрахування на соціальні заходи, електроенергію та накладні витрати. Отримані результати дозволяють визначити повну собівартість розробки та оцінити економічну ефективність створеного програмного забезпечення.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						56
Змн.	Арк	№ докум.	Підпис	Дата		

Таблиця 4.3 – Кошторис витрат на розробку IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32

№	Зміст витрат	Сума, грн.	В % до загальної суми
	Витрати на оплату праці (основну і додаткову заробітну плату)	78 062,00	61,2
	Єдиний соціальний внесок	17 173,64	13,5
	Витрати на електроенергію	1 185,94	1,0
	Амортизаційні відрахування	3 472,00	2,9
	Накладні витрати	27 321,70	21,4
	Собівартість	127 215,28	100

Повна собівартість проєкту (Св) розраховується як сума всіх статей витрат за формулою:

$$C_{\text{в}} = C_{\text{о.п}} + C_{\text{с.з}} + C_{\text{е}} + A + H_{\text{в}} \quad (4.8)$$

$$C_{\text{в}} = 127\,215,28 \text{ грн.}$$

4.7 Розрахунок ціни НДР

Розрахунок ціни науково-дослідної роботи (НДР), у даному випадку — розробки програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32, здійснюється з урахуванням повної собівартості робіт та запланованого прибутку. Ціна розробки повинна забезпечувати відшкодування всіх понесених витрат і формування прибутку підприємства.

Ціна НДР визначається за формулою:

$$Ц = C_{\text{в}} * (1 + P_{\text{рен}}) * (1 + \text{ПДВ}), \quad (4.9)$$

де: Ц – кінцева ціна проєкту, грн.; Св – собівартість; Ррен. – рівень рентабельності, 30 %; ПДВ – ставка податку на додану вартість, 20 %

$$Ц = 127\,215,28 * (1 + 0,30) * (1 + 0,20) = 198\,455,84 \text{ грн}$$

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						57
Змн.	Арк	№ докум.	Підпис	Дата		

4.8. Визначення економічної ефективності і терміну окупності капітальних вкладень

Ефективність виробництва — це комплексний показник, що відображає кінцеві результати використання всіх ресурсів (робочої сили, основних фондів, матеріальних ресурсів) за певний проміжок часу. Для оцінки економічної ефективності проекту розрахуємо ключові показники: прибуток, чисту теперішню вартість (ЧТВ) та термін окупності (Ток).

Чиста теперішня вартість (Net Present Value, NPV) є одним із основних показників оцінки ефективності інвестиційного проекту. Вона характеризує різницю між дисконтованими грошовими надходженнями та витратами.

$$\text{ЧТВ} = -\text{Св} + \sum_{i=1}^t \frac{\Gamma_{\Pi}}{(1+i)^t}, \quad (4.11)$$

де: Св – собівартість розробки; Γ_{Π} – грошовий потік за t -ий рік; t – відповідний рік проекту; i – ставка дисконтування.

$$\text{ЧТВ} = -127\,215,28 + \frac{71\,240,56}{(1+0,1)} + \frac{71\,240,56}{(1+0,1)^2} + \frac{71\,240,56}{(1+0,1)^3} = 49\,949,00$$

Оскільки $\text{ЧТВ} > 0$, проект є економічно вигідним.

3) Розрахунок терміну окупності (Ток)

Термін окупності визначається за формулою: (4.12)

$$T_{\text{ок}} = T_{\text{пв}} + \frac{НВ}{\Gamma_{\text{пр}}} \quad (4.12)$$

де: $T_{\text{пв}}$ – період до повного відшкодування витрат, років; $НВ$ – невідшкодовані витрати на початок року, грн.; $\Gamma_{\text{пр}}$ – грошовий потік на початок року, грн.

$$T_{\text{ок}} = 2 + \frac{3574,8}{71\,240,56} = 2,1$$

На основі проведених розрахунків складемо фінальну таблицю техніко-економічних показників.

					2026.КВР.123.405.04.00.00 ПЗ	Арк.
						58
Змн.	Арк	№ докум.	Підпис	Дата		

Таблиця 4.5 – Техніко-економічні показники

№ п/п	Показник	Значення
1.	Собівартість, грн.	127 215,28
2.	Плановий прибуток або грошовий потік, грн.	71 240,56
3.	Ціна, грн.	198 455,84
4.	Чиста теперішня вартість, грн.	49 949,00
5.	Термін окупності, рік	2,1

Таким чином, розробка програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32 є економічно доцільною. Отриманий прибуток забезпечує повернення вкладених коштів менш ніж за два роки, що свідчить про ефективність та перспективність впровадження даної системи.

5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА ЖИТТІДІЯЛЬНОСТІ

Під час розроблення та експлуатації низьковольтних інтелектуальних систем необхідно дотримуватися вимог Закону України «Про охорону праці», а також чинних правил технічної експлуатації електроустановок споживачів. Ці документи встановлюють загальні організаційні й технічні вимоги безпечної роботи електроустановок, навчання персоналу та проведення періодичних перевірок знань.

5.1 Вплив параметрів мікроклімату на організм людини

Мікроклімат виробничих і побутових приміщень є сукупністю фізичних параметрів повітряного середовища, які безпосередньо впливають на тепловий стан організму людини, її самопочуття, працездатність та рівень безпеки. До основних параметрів мікроклімату належать температура повітря, відносна вологість, швидкість руху повітря, а також концентрація шкідливих домішок і газів. У контексті сучасних інтелектуальних систем особливу увагу також приділяють показникам якості повітря, зокрема концентрації CO₂, летких органічних сполук (VOC), пилових частинок та інших забруднювачів.

У межах даної роботи, яка присвячена розробці IoT-системи моніторингу якості повітря та автоматичного керування вентиляцією на базі ESP32, параметри мікроклімату розглядаються як об'єкти постійного контролю за допомогою датчиків і програмних алгоритмів. Такий підхід дозволяє не лише фіксувати поточний стан повітря, але й автоматично реагувати на його зміну шляхом керування вентиляційними пристроями.

5.1.1. Температура повітря та її вплив

Температура є одним із ключових параметрів мікроклімату, оскільки вона визначає інтенсивність теплообміну між організмом людини та навколишнім середовищем. За підвищеної температури повітря відбувається перевантаження

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						60
Змн.	Арк	№ докум.	Підпис	Дата		

терморегуляційної системи організму, що може призводити до перегрівання, зниження концентрації уваги, швидкої втомлюваності та зростання ризику помилок під час виконання робіт.

Надто низька температура, навпаки, викликає переохолодження, спазм судин та зниження працездатності. У виробничих умовах це особливо небезпечно для працівників, які працюють з електронними системами, оскільки зниження концентрації може призвести до помилок у монтажі або експлуатації обладнання.

У IoT-системі на базі ESP32 вимірювання температури може здійснюватися за допомогою цифрових або аналогових датчиків DHT22, а отримані дані використовуються для прийняття рішення щодо вмикання або регулювання вентиляції.

5.1.2. Відносна вологість повітря

Вологість повітря є важливим параметром, який впливає на тепловідчуття людини та загальний комфорт. При підвищеній вологості ускладнюється випаровування поту, що призводить до перегрівання організму навіть при відносно помірній температурі. При надто низькій вологості спостерігається пересихання слизових оболонок, подразнення дихальних шляхів та зниження захисних функцій організму.

Для підтримання оптимального рівня вологості в сучасних системах застосовуються автоматизовані системи вентиляції та зволоження/осушення повітря. У межах розроблюваної IoT-системи ESP32 може отримувати дані про вологість з датчиків і передавати їх у програмний модуль керування вентиляцією, який приймає рішення щодо інтенсивності повітрообміну.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						61
Змн.	Арк	№ докум.	Підпис	Дата		

5.1.3. Якість повітря та газові домішки

Окрему роль у формуванні мікроклімату відіграє якість повітря, яка визначається наявністю шкідливих газів і домішок. До них належать вуглекислий газ (CO₂), чадний газ (CO), леткі органічні сполуки (VOC), а також інші забруднювачі, що можуть накопичуватися в закритих приміщеннях.

Підвищена концентрація CO₂ є одним із найбільш поширених факторів погіршення самопочуття. Вона викликає сонливість, головний біль, зниження концентрації та загальну втоми. У виробничих або навчальних приміщеннях це безпосередньо впливає на ефективність роботи та якість виконання завдань.

У даному проєкті для моніторингу якості повітря може використовуватися газовий датчик типу MQ-135, який дозволяє оцінювати рівень забруднення повітря за сукупністю газів. ESP32 обробляє отримані дані та передає їх у систему керування вентиляцією, яка може автоматично вмикати вентилятор або змінювати режим його роботи.

5.1.4. Рух повітря та вентиляції

Швидкість руху повітря є важливим фактором, що впливає на тепловіддачу організму. Занадто слабкий повітрообмін призводить до застою повітря, накопичення тепла та забруднень, тоді як надмірний рух повітря може викликати дискомфорт і локальне переохолодження.

Саме тому вентиляція є ключовим елементом забезпечення нормального мікроклімату. Вона виконує дві основні функції: подачу свіжого повітря та видалення забрудненого. У сучасних інтелектуальних системах вентиляція все частіше стає автоматизованою та адаптивною, тобто її робота залежить від даних сенсорів у реальному часі.

У межах даної роботи використовується принцип автоматичного керування вентиляцією, при якому ESP32 аналізує параметри мікроклімату та керує реле, що вмикає або вимикає вентилятор. Це дозволяє підтримувати оптимальні умови без постійного втручання людини.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						62
Змн.	Арк	№ докум.	Підпис	Дата		

5.2 Вентиляція виробничих приміщень

Вентиляція виробничих приміщень є одним із основних технічних засобів забезпечення безпечних і комфортних умов праці. Її призначення полягає у підтриманні нормованих параметрів повітряного середовища шляхом подачі свіжого повітря, видалення забрудненого повітря, зменшення надлишкового тепла, вологі та шкідливих домішок. Чинні правила з охорони праці та безпечної експлуатації систем вентиляції встановлюють вимоги до їхнього влаштування, експлуатації та обслуговування, щоб система працювала стабільно та не створювала додаткових небезпек для персоналу.

У контексті даної роботи, присвяченої розробці програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32, вентиляція розглядається не лише як інженерна система будівлі, а як керований об'єкт автоматизації. Це означає, що її робота може залежати від даних датчиків температури, вологості, концентрації забруднювачів і якості повітря. Такий підхід дає змогу вмикати або регулювати вентиляцію тільки тоді, коли показники середовища справді виходять за межі допустимих значень, тобто забезпечувати енергоефективність і комфорт одночасно. Це узгоджується з вимогами до безпечної експлуатації інженерних систем та загальними принципами охорони праці.

Ефективність вентиляції безпосередньо пов'язана з мікрокліматом приміщення. Мікроклімат визначається температурою повітря, відносною вологістю, швидкістю руху повітря, температурою навколишніх поверхонь і тепловим випромінюванням. Саме ці фактори впливають на тепловий обмін людини з навколишнім середовищем, а отже — на самопочуття, працездатність і рівень втоми. Якщо повітрообмін недостатній, у приміщенні накопичуються тепло, волога та домішки, що погіршує умови праці. Якщо ж повітряний потік надто інтенсивний, це може спричинити дискомфорт і локальне переохолодження. Отже, вентиляція повинна бути не просто ввімкнена, а налаштована так, щоб підтримувати збалансовані параметри середовища.

Для виробничих приміщень найчастіше застосовують природну, механічну та

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						63
Змн.	Арк	№ докум.	Підпис	Дата		

змішану вентиляцію. Природна вентиляція базується на різниці температур і тиску повітря та може реалізовуватися через кватирки, фрамуги, вентиляційні канали й інші отвори. Її перевагою є простота, проте вона суттєво залежить від погодних умов і не завжди забезпечує стабільний результат. Механічна вентиляція використовує вентилятори, повітропроводи, решітки, фільтри та інше обладнання, тому дозволяє точніше керувати обміном повітря. Саме механічна схема є найбільш придатною для автоматизованого керування за допомогою ESP32, оскільки електронний контролер може вмикати вентилятор, змінювати режим його роботи або передавати дані користувачу в режимі реального часу.

У сучасних умовах особливого значення набуває контроль не лише температури, а й складу повітря. У закритих приміщеннях без достатнього повітрообміну швидко зростає концентрація вуглекислого газу, летких органічних сполук та інших домішок, що негативно впливають на концентрацію уваги, викликають втоми і знижують продуктивність праці. Саме тому системи вентиляції мають працювати у зв'язці з датчиками контролю повітря. Для IoT-рішення на базі ESP32 це особливо зручно: мікроконтролер збирає дані, виконує їх обробку та за заданим алгоритмом подає команду на реле або інший виконавчий пристрій. У результаті формується замкнений цикл керування, де стан повітря безпосередньо впливає на роботу вентилятора.

Під час проектування вентиляції необхідно враховувати вимоги до безпечної експлуатації. Нормативні документи встановлюють, що системи опалення, вентиляції та кондиціонування виробничих приміщень мають відповідати чинному законодавству, а їх експлуатація повинна здійснюватися з дотриманням правил охорони праці. Це стосується не лише самого обладнання, а й порядку його технічного обслуговування, доступу до рухомих частин, огляду електричних з'єднань, контролю стану фільтрів, каналів і повітроводів. Порушення цих вимог може призвести до зниження ефективності вентиляції, поширення пилу та забруднень, а також до аварійних ситуацій.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						64
Змн.	Арк	№ докум.	Підпис	Дата		

Важливим питанням є також пожежна безпека. Під час експлуатації систем вентиляції та кондиціонування не допускається недбале поводження з елементами, що можуть впливати на поширення диму або вогню, а вентиляційне обладнання повинно працювати узгоджено з протипожежними заходами. У виробничих приміщеннях доцільно передбачати технічні засоби протипожежного захисту, а за потреби — системи димовидалення та відповідні датчики контролю. Для інтелектуальної системи на базі ESP32 це означає можливість доповнення основного алгоритму окремими сценаріями безпеки, наприклад примусовим увімкненням вентиляції або аварійним вимкненням окремих пристроїв у разі виявлення небезпечної ситуації.

З точки зору енергозбереження та екологічності автоматизована вентиляція має суттєві переваги перед постійно ввімкненими системами. Коли контроль здійснюється на основі сенсорних даних, вентилятор працює тільки за потреби, що зменшує споживання електроенергії та знижує зношення обладнання. Крім того, зменшується надмірний шум і підвищується загальний комфорт у приміщенні. Для IoT-системи це особливо актуально, оскільки програмне забезпечення ESP32 може реалізовувати порогове або адаптивне керування з урахуванням декількох параметрів одночасно: температури, вологості, рівня забруднення повітря та часу роботи вентиляції. Таким чином, вентиляція стає не просто допоміжною функцією, а частиною інтелектуальної системи керування мікрокліматом.

Отже, вентиляція виробничих приміщень є обов'язковою складовою забезпечення безпечних умов праці. Вона впливає на мікроклімат, стан здоров'я працівників, якість повітря та надійність функціонування обладнання. Для теми даної роботи вентиляція має особливе значення, оскільки саме на її основі реалізується автоматичне керування через ESP32. Використання датчиків і програмних алгоритмів дозволяє створити систему, яка своєчасно реагує на зміну стану повітря, підвищує комфорт користувача, знижує енергоспоживання та сприяє виконанню вимог охорони праці.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						65
Змн.	Арк	№ докум.	Підпис	Дата		

5.3 Організація безпечної експлуатації низьковольтних інтелектуальних систем

Безпечна експлуатація низьковольтних інтелектуальних систем є комплексом організаційних і технічних заходів, спрямованих на запобігання ураженню електричним струмом, виникненню аварійних ситуацій, виходу з ладу обладнання та негативному впливу на навколишнє середовище. У контексті даної роботи, що присвячена розробці програмного забезпечення IoT-системи моніторингу якості повітря та керування вентиляцією на базі ESP32, питання безпеки набувають особливого значення, оскільки система поєднує електронні компоненти, датчики, виконавчі пристрої та програмне керування в єдину автоматизовану структуру.

Основними нормативними документами, що регламентують вимоги до безпечної експлуатації електроустановок, є Закон України «Про охорону праці» та Правила технічної експлуатації електроустановок споживачів. Вони визначають загальні принципи організації безпечної роботи, вимоги до персоналу, порядок технічного обслуговування та умови використання електрообладнання. Дотримання цих вимог є обов'язковим незалежно від складності системи або її призначення.

5.3.1. Електробезпека при експлуатації IoT-систем

Низьковольтні системи, такі як ESP32-платформи, зазвичай працюють із напругою 3.3 В або 5 В, однак у складі системи можуть використовуватися додаткові елементи, зокрема блоки живлення, реле, вентилятори або інші виконавчі пристрої, які підключаються до мережі змінного струму 220 В. Саме ці ділянки системи становлять підвищену небезпеку та потребують особливої уваги.

Для забезпечення електробезпеки необхідно:

- використовувати справні та сертифіковані блоки живлення із захистом від короткого замикання та перевантаження;
- забезпечувати гальванічну розв'язку між низьковольтною частиною (ESP32) та силовими колами;

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						66
Змн.	Арк	№ докум.	Підпис	Дата		

- застосовувати реле або твердотільні ключі з відповідним запасом за струмом і напругою;
- уникати відкритих струмопровідних частин у корпусі пристрою;
- використовувати ізольовані дроти та надійні з'єднання.

Особливу увагу слід приділяти монтажу системи, оскільки неправильне підключення може призвести до короткого замикання, перегріву компонентів або виходу з ладу мікроконтролера ESP32.

5.3.2. Захист апаратної частини системи

Апаратна частина IoT-системи повинна бути захищена від впливу зовнішніх факторів, таких як пил, волога, перепади температури та електромагнітні завади. Для цього рекомендується розміщувати електронні компоненти в захисних корпусах із достатнім рівнем вентиляції.

Важливими заходами є:

- встановлення запобіжників у ланцюгах живлення;
- використання стабілізаторів напруги для захисту від перепадів;
- застосування захисту від перенапруг (TVS-діоди, фільтри);
- правильне заземлення металевих корпусів (за наявності).

У системах керування вентиляцією особливу роль відіграє реле, яке комутує навантаження. Воно повинно мати запас по струму та бути розраховане на тривалу роботу в умовах циклічного вмикання та вимикання вентилятора.

5.3.3. Програмна безпека та надійність роботи

Окрім апаратної частини, важливим аспектом є програмна безпека системи. Оскільки ESP32 виконує функції збору даних із датчиків та керування виконавчими пристроями, помилки в алгоритмах можуть призвести до некоректної роботи вентиляції або аварійних ситуацій.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						67
Змн.	Арк	№ докум.	Підпис	Дата		

Для підвищення надійності програмного забезпечення доцільно:

- передбачати перевірку коректності даних із датчиків (фільтрація шумів і вибірових помилок);
- реалізовувати захисні таймери, які запобігають безперервній роботі вентилятора без потреби;
- використовувати логіку аварійного режиму (fail-safe), коли при збоях система переходить у безпечний стан;
- передбачати відновлення роботи після перезавантаження (watchdog timer);
- уникати перевантаження мікроконтролера зайвими задачами.

У випадку IoT-систем важливим є також стабільність зв'язку між ESP32 та сервером або веб-інтерфейсом. При втраті зв'язку система повинна продовжувати працювати автономно на основі локальних даних із датчиків.

5.3.4. Організація технічного обслуговування

Безпечна експлуатація неможлива без регулярного технічного обслуговування системи. Воно включає перевірку стану електричних з'єднань, очищення датчиків від пилу, перевірку працездатності вентилятора та контроль стабільності живлення. Рекомендується періодично виконувати:

- огляд контактів і клем на предмет окислення або послаблення;
- перевірку роботи реле та виконавчих пристроїв;
- калібрування датчиків якості повітря та температури;
- тестування аварійних режимів системи;
- оновлення програмного забезпечення при необхідності.

Такі заходи дозволяють підтримувати стабільну роботу системи та зменшують ризик несподіваних відмов.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						68
Змн.	Арк	№ докум.	Підпис	Дата		

5.3.5. Екологічні та експлуатаційні аспекти

IoT-системи моніторингу та керування вентиляцією мають також позитивний екологічний ефект. Завдяки автоматичному керуванню вентиляцією зменшується надмірне споживання електроенергії, оскільки пристрої працюють лише тоді, коли це необхідно. Це дозволяє знизити енергетичні витрати та підвищити загальну ефективність системи.

Крім того, використання електронних компонентів потребує врахування питань утилізації. Після завершення терміну експлуатації датчики, плати та інші елементи повинні утилізуватися відповідно до вимог поводження з електронними відходами, щоб зменшити негативний вплив на довкілля.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи було розроблено програмне забезпечення IoT-системи моніторингу повітря та керування вентиляцією на базі ESP32. Проведено аналіз сучасних технологій Інтернету речей та засобів контролю параметрів повітряного середовища, що дозволило обрати оптимальні програмні та апаратні засоби для реалізації системи.

У ході роботи було розроблено структуру програмно-апаратного комплексу, алгоритм функціонування системи та програмне забезпечення для збору й обробки даних від датчиків температури, вологості та якості повітря. Реалізовано відображення інформації на LCD-дисплеї та передачу даних до хмарної платформи Blynk IoT для віддаленого моніторингу.

Проведене тестування підтвердило коректність роботи програмного забезпечення, стабільність передачі даних та працездатність системи в реальному часі. Розроблений програмний комплекс може бути використаний для контролю мікроклімату в житлових, навчальних та офісних приміщеннях, а також може бути розширений додатковими функціями автоматизованого керування вентиляційними системами.

Поставлену мету роботи досягнуто, а всі визначені завдання виконано в повному обсязі.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						70
Змн.	Арк	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ

1. Буров Є.В. Комп'ютерні інформаційні системи. – Львів: Новий Світ-2000, 2020. – 412 с.
2. Войтко В.В. Архітектура комп'ютерів та мікропроцесорні системи. – Вінниця: ВНТУ, 2018. – 245 с.
3. ГОСТ 19.701-90. Схеми алгоритмів, програм, даних і систем. Умовні позначення та правила виконання. – Москва: Держстандарт СРСР, 1990. – 26 с.
4. Грицюк Ю.І. Основи програмування та алгоритмічні мови. – Львів: Новий Світ-2000, 2019. – 304 с.
5. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлення. – Київ: ДП «УкрНДНЦ», 2016. – 31 с.
6. Жуйков В.Я., Терещенко Т.О. Мікропроцесорні системи та пристрої. – Київ: КПІ ім. Ігоря Сікорського, 2018. – 356 с.
7. Кадах В.М. Інтернет речей: технології та застосування. – Київ: КПІ ім. Ігоря Сікорського, 2021. – 278 с.
8. Конахович Г.Ф., Пузиренко О.Ю. Комп'ютерні мережі та телекомунікації. – Київ: Каравела, 2021. – 344 с.
9. Кравець П.О. Проектування інформаційних систем. – Львів: Видавництво Львівської політехніки, 2020. – 318 с.
10. Організація баз даних та знань: навчальний посібник / за ред. В.В. Пасічника. – Львів: Магнолія 2006, 2020. – 440 с.
11. Основи Інтернету речей: навчальний посібник / за ред. В.М. Глушкова. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 290 с.
12. Програмна інженерія: підручник / за ред. С.А. Бешелева. – Київ: Ліра-К, 2020. – 512 с.
13. Соммервілл І. Інженерія програмного забезпечення. – Київ: Вільямс, 2021. – 824 с.
14. Таненбаум Е., Бос Г. Комп'ютерні мережі. – Київ: Вільямс, 2020. – 960 с.
15. Aosong Electronics Co., Ltd. DHT11 Temperature and Humidity Sensor Datasheet. – Technical Documentation, 2023. – 10 p.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						71
Змн.	Арк	№ докум.	Підпис	Дата		

16. Arduino Documentation. Arduino IDE User Guide. URL: <https://docs.arduino.cc/software/ide/> (дата звернення: 21.06.2026).
17. Blynk IoT Platform Documentation. URL: <https://docs.blynk.io/en> (дата звернення: 21.06.2026).
18. Espressif Systems. ESP32-C6 Series Datasheet. – Espressif Systems, 2024. – 156 p.
19. MQ135 Air Quality Sensor Datasheet. Zhengzhou Winsen Electronics Technology Co., Ltd. – Technical Documentation, 2023. – 12 p.
20. Seeed Studio. XIAO ESP32C6 Getting Started Guide. – Seeed Studio, 2024. – 48 p.

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						72
Змн.	Арк	№ докум.	Підпис	Дата		

ДОДАТОК

Код програми:

```
#define BLYNK_TEMPLATE_ID "Your_Temp_ID"
#define BLYNK_TEMPLATE_NAME "Air Quality Monitoring"
#define BLYNK_AUTH_TOKEN "Your_Auth-Token"

#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>

#include <DHT.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

byte degree_symbol[8] =
{
0b00111,
0b00101,
0b00111,
0b00000,
0b00000,
0b00000,
0b00000,
0b00000
};

char auth[] = BLYNK_AUTH_TOKEN;

char ssid[] = "WiFi name";
```

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						73
Змн.	Арк	№ докум.	Підпис	Дата		


```

char pass[] = "WiFi Password";

BlynkTimer timer;

#define gas A0
int sensorThreshold = 100;

#define DHTPIN D2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

void sendSensor()
{

float h = dht.readHumidity();
float t = dht.readTemperature();

if (isnan(h) || isnan(t)) {
Serial.println("Failed to read from DHT sensor!");
return;
}

int analogSensor = analogRead(gas);
Blynk.virtualWrite(V2, analogSensor);
Serial.print("Gas Value: ");
Serial.println(analogSensor);
Blynk.virtualWrite(V0, t);

```

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						74
Змн.	Арк	№ докум.	Підпис	Дата		

```

Blynk.virtualWrite(V1, h);

Serial.print("Temperature : ");
Serial.print(t);
Serial.print(" Humidity : ");
Serial.println(h);

}

void setup()
{

Serial.begin(115200);

Blynk.begin(auth, ssid, pass);
dht.begin();
timer.setInterval(30000L, sendSensor);

lcd.begin();

lcd.setCursor(3,0);
lcd.print(" Air Quality");
lcd.setCursor(3,1);
lcd.print("Monitoring");
delay(2000);
lcd.clear();

```

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						75
Змн.	Арк	№ докум.	Підпис	Дата		

```

lcd.setCursor(3,0);
lcd.print("Air Quality");
lcd.setCursor(3,1);
lcd.print("Monitoring");
delay(2000);
lcd.clear();
}

void loop()
{
  Blynk.run();
  timer.run();
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  int gasValue = analogRead(gas);
  lcd.setCursor(0,0);
  lcd.print("Temperature ");
  lcd.setCursor(0,1);
  lcd.print(t);
  lcd.setCursor(6,1);
  lcd.write(1);
  lcd.createChar(1, degree_symbol);
  lcd.setCursor(7,1);
  lcd.print("C");
  delay(4000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Humidity
");

```

```

lcd.print(h);
lcd.print("%");
delay(4000);
lcd.clear();

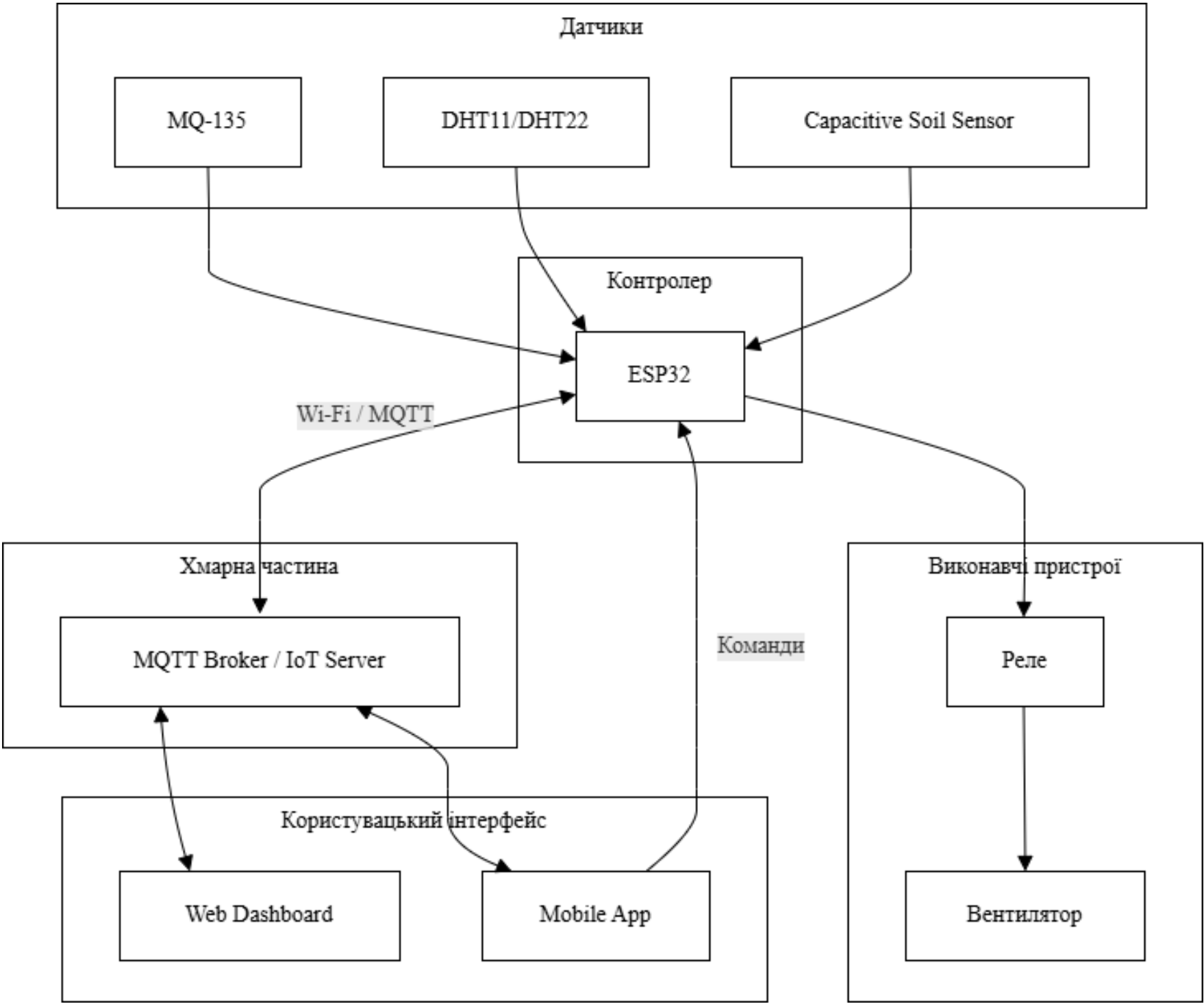
Serial.println("Gas
Value");
Serial.println(gasValue);
if(gasValue<1200)
{
lcd.setCursor(0,0);
lcd.print("Gas Value: ");
lcd.print(gasValue);
lcd.setCursor(0, 1);
lcd.print("Fresh Air");
Serial.println("Fresh
Air"); delay(4000);
lcd.clear();
}
else if(gasValue>1200)
{
lcd.setCursor(0,0);
lcd.print(gasValue);
lcd.setCursor(0, 1);
lcd.print("Bad Air");
Serial.println("Bad
Air"); delay(4000);
lcd.clear();
}

```

					2026.KBP.123.405.04.00.00 ПЗ	Арк.
						77
Змн.	Арк	№ докум.	Підпис	Дата		

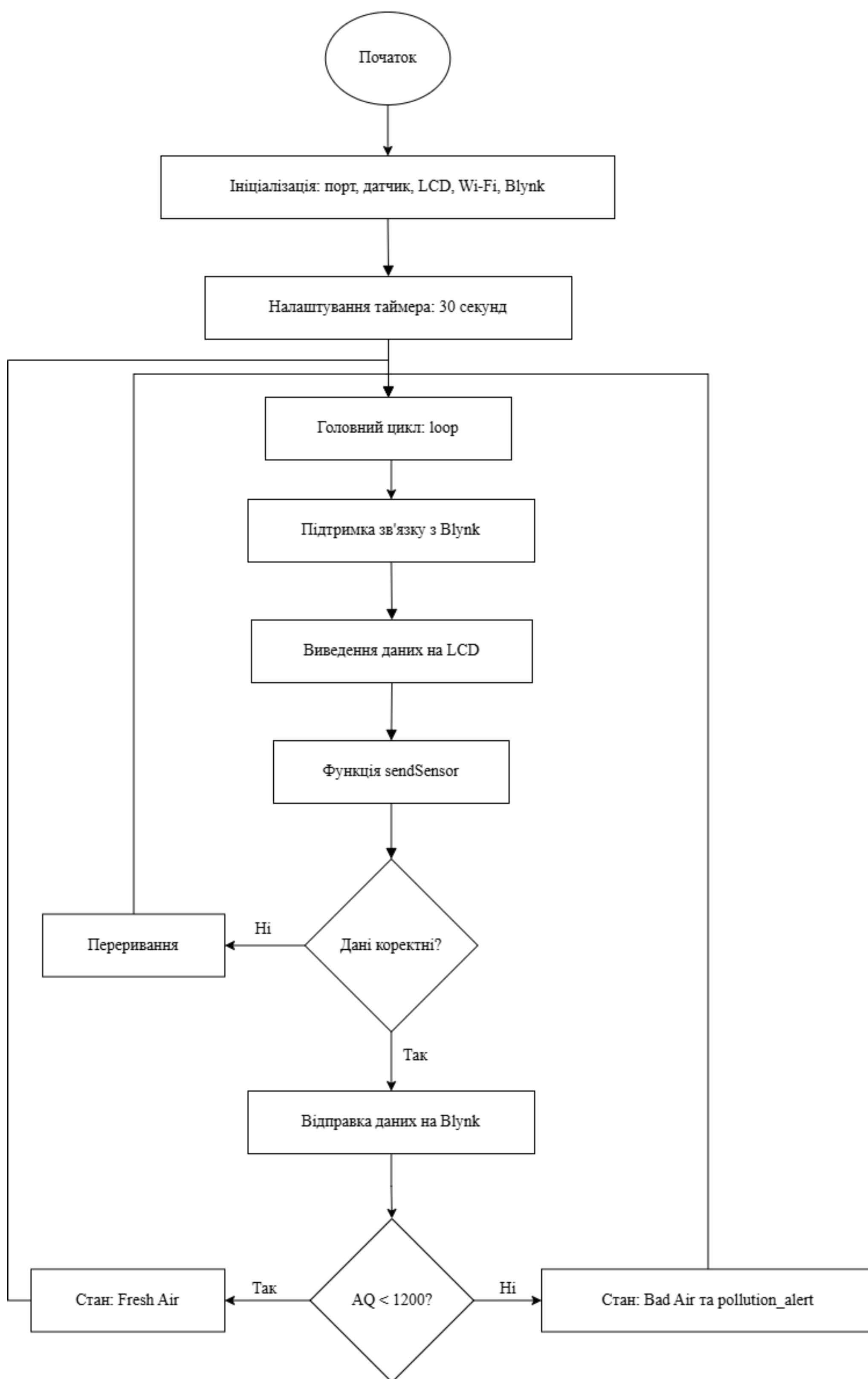
```
if(gasValue > 1200){  
  Blynk.logEvent("pollution_alert","Bad Air");  
}  
}
```

Структурна схема IoT-системи



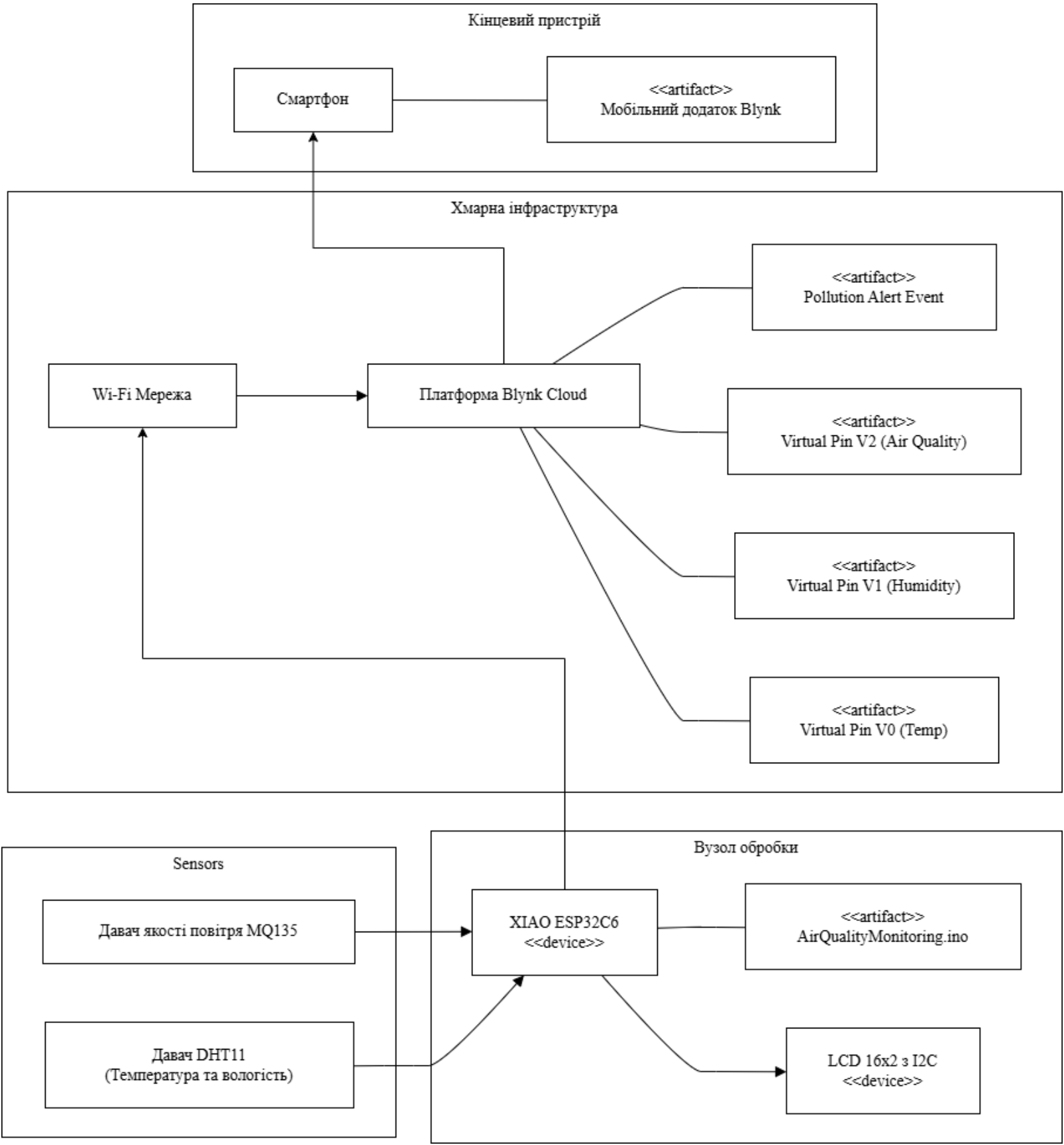
						2026.KBP.123.405.04.00.00TB					
						Розробка програмного забезпечення IoT-системи моніторингу повітря та керування вентиляцією на базі ESP32			Літ.	Арх.	Архув
Зм. Арк.	№ докум.	Титул	Дата			Зм. Арк.			Літ.	Арх.	Архув
Розроб.		Соломаш Р.А.									
Перевірив											
Реценз.											
А. Контр.	Юлія А.В.										
Рис.											
Структурна схема IoT-системи						ВСП ТФК ТНТУ КІ-405					

Блок-схема роботи програмного забезпечення



						2026.KBP.123.405.04.00.00 БС					
						Розробка програмного забезпечення ІoT-системи моніторингу повітря та керування вентиляцією на базі ESP32	Літ.	Маса	Масштаб		
							Аркуш	Аркуші	1		
Зм.	Арк.	№ докум.	Підпис	Дата		Блок-схема роботи програмного забезпечення	ВСП «ТФК ТНТУ»,				
Розроб.	Вівчар В.В.						KI-405				
Перевір.	Болюбаш Р.А.										
Консульта.											
Реценз.											
Н.контр.	Юзків А. В.										
Затв.											

Діаграма розгортання



						2026.KBP.123.405.04.00.00 CC					
МЗ Арк.	№ докум.	Підп.	Дата	Розробка програмного забезпечення IoT-системи моніторингу повітря та керування вентиляцією на базі ESP32					Літ.	Арк.	Аркушів
Розроб.	Савчук В.В.										
Перевірив	Колодій Р.А.										
Рецензент									Арк.		Аркушів
Н. контрол.	Одзюк А.В.	Діаграма розгортання (Deployment Diagram)					ВСП ТФК ТНТУ				
Уні.											

Таблиця технічно-економічних показників

№ п.п.	Назва показників	Одиниці вимірювання	Значення	№ п.п.	Назва показників	Одиниці вимірювання	Значення
1	Тип системи	-	IoT-система моніторингу повітря	6	Використаний мікроконтролер	-	ESP32 (XIAO ESP32C6)
2	Кількість датчиків	шт	2	7	Тип датчиків	-	DHT11, MQ135
3	Кількість виконавчих пристроїв	шт	1	8	Тип зв'язку	-	Wi-Fi
4	Тривалість виконання НДР	год	186	9	Платформа моніторингу	-	Blynk IoT
5	Собівартість розробки	грн	142648,30	10	Термін окупності	роки	1,79

						2026.KBP.123.405.04.00.00ТБ					
						Розробка програмного забезпечення IoT-системи моніторингу повітря та			Лист	Арк.	Аркушів
ЗМ	Арк.	№ докум.	Підп.	Дата							
Розроб.		Венер В.В.									
Перевіряв		Болобаш Р.А.				керування вентиляцією на базі ESP32					
Реценз.											
									Арк.	Аркушів 1	
Н. Контр.		Юшків А.В.				Таблиця технічно-економічних показників			ВСП «Технічні» ТНТУ		
Утв.									KI-405п		

