Форма № 24

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Комп'ютерних наук і технологій
(повне найменування факультету)

Комп'ютерні системи та мережі
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)
бакалаврський
(ступінь вищої освіти)

на тему: ПРОЄКТУВАННЯ МЕРЕЖІ ПЕРЕДАЧІ ДАНИХ КОТЕДЖНОГО КОМПЛЕКСУ ІЗ ЗАСТОСУВАННЯМ СТАНДАРТІВ СЕНСОРНИХ МЕРЕЖ

Виконав(ла): студент(ка) 4 курсу,
групи КНТ-512сп

Спеціальності
123 Комп'ютерна інженерія
(код і найменування спеціальності)

Освітня програма (спеціалізація)
Комп'ютерна інженерія
(назва освітньої програми (спеціалізації)

ПІТЮРЕНКО Н.К.
(ПРІЗВИЩЕ та ініціали)

Керівник КИРИЧЕК Г.Г.
(ПРІЗВИЩЕ та ініціали)
Рецензент КОЗІНА Г.Л.
(ПРІЗВИЩЕ та ініціали)

2025

Форма № 25

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет Комп'ютерних наук і технологій
Кафедра комп'ютерних систем та мереж
Ступінь вищої освіти бакалаврський
Спеціальність 123 Комп'ютерна інженерія
(код і найменування)
Освітня програма (спеціалізація) Комп'ютерна інженерія
(назва освітньої програми (спеціалізації)

ЗАТВЕРДЖУЮ
Завідувач кафедри КУДЕРМЕТОВ Р.К.

«14» квітня 2025 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Пітюренко Нікіти Костянтиновича
(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Проєктування мережі передачі даних котеджного комплексу із застосуванням стандартів сенсорних мереж

керівник проєкту (роботи) к.т.н., доцент, КИРИЧЕК Г.Г.,
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)
затверджені наказом закладу вищої освіти від « 08 » квітня 2025 року № 151

2. Строк подання студентом проєкту (роботи) 01.06.2025 р.

3. Вихідні дані до проєкту (роботи) Сенсорна мережа котеджного комплексу на 60 керованих комутаторів із використанням технології Ethernet стандарту 1000BaseFX, логічного поділу мережі на VLAN, списків контролю доступу (ACL), 4 бездротових точок доступу та технології Wi-Fi стандарту 802.11ax, динамічної адресації IPv4 та статичної маршрутизації. Інтеграція сенсорних пристроїв (камери, детектори, кнопки сповіщення), централізоване керування через WLC-контролери та сервер IoT, а також резервування каналів між віддаленими сегментами.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1) Опис предметної області;
технічне завдання та вимоги до мережі, проєктування мережі передачі даних, налаштування роботи сенсорної мережі та моніторинг її стану

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)
Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРІЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-3	КИРИЧЕК Г.Г.		
Нормоконтроль	ЩЕРБАК Н.В.		

7. Дата видачі завдання «14» лютого 2025 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз предметної області	до 18.04.2025	
2	Визначення та аналіз вимог до програмного	до 22.04.2025	
	забезпечення		
3	Розробка специфікації вимог SRS	до 24.04.2025	
4	Розробка діаграми варіантів використання	до 28.04.2025	
5	Розробка діаграми послідовності	до 01.05.2025	
6	Розробка описів прецедентів	до 05.05.2025	
7	Проєктування архітектури програмного	до 12.05.2025	
	забезпечення		
8	Проєктування інтерфейсу користувача	до 22.05.2025	
9	Оформлення пояснювальної записки	до 25.05.2025	
10	Проходження нормоконтролю	до 01.06.2025	
11	Перевірка на наявність академічного плагіату	до 03.06.2025	
12	Проходження рецензування	до 10.06.2025	

Студент(ка) _____ Нікіта ПІТЮРЕНКО
(підпис) (Ім'я ПРІЗВИЩЕ)

Керівник проєкту (роботи) _____ Галина КИРИЧЕК
(підпис) (Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 79 с., 5 табл., 26 рис., 2 додатки, 21 джерело.

CISCO PACKET TRACER, IP-АДРЕСАЦІЯ, DHCP, DNS, ACL, FRAME RELAY, 802.11AX, 1000BASEFX, СТАТИЧНА МАРШРУТИЗАЦІЯ, КОМУТАЦІЯ, ЦЕНТРАЛІЗОВАНЕ КЕРУВАННЯ.

Метою роботи є розробка макету мережевої інфраструктури котеджного комплексу, моделювання та налаштування бездротового покриття, а також реалізація програмного забезпечення для аналізу та прогнозування стану мережі.

Об’єкт розробки – проєкт мережі передачі даних для котеджного комплексу з інтеграцією сенсорних технологій та системи прогнозування стану мережі.

Предметом дослідження є методи проєктування, налаштування та інтелектуального моніторингу мережевої інфраструктури котеджного комплексу з використанням сенсорних технологій.

Проєкт складається з трьох розділів.

Перший розділ містить аналіз технологій побудови мереж для котеджних комплексів, огляд стандартів сенсорних мереж, дослідження існуючих рішень та формування технічних вимог.

Другий розділ присвячено розробці схеми мережі, налаштуванню її компонентів, інтеграції IoT-пристроїв та тестуванню працездатності, включаючи доступ до інтернету, аварійне сповіщення та сенсорні механізми.

У третьому розділі висвітлено налаштування сенсорної мережі, реалізацію відеоспостереження, аварійного сповіщення, NAT-доступу до Інтернету та модулю прогнозування стану мережі; тестування підтвердило її працездатність і готовність до використання.

ABSTRACT

Explanatory note to the bachelor's qualification thesis:
79 pages, 5 tables, 26 figures, 2 appendices, 21 references.

CISCO PACKET TRACER, IP ADDRESSING, DHCP, DNS, ACL, FRAME RELAY, 802.11AX, 1000BASEFX, STATIC ROUTING, SWITCHING, CENTRALIZED MANAGEMENT.

The aim of the work is to develop a model of the network infrastructure for a cottage complex, simulate and configure wireless coverage, and implement software for analyzing and predicting the network state.

The object of development is a data transmission network project for a cottage complex with the integration of sensor technologies and a system for predicting the network state.

The subject of research is the methods of designing, configuring, and intelligently monitoring the network infrastructure of a cottage complex using sensor technologies.

The project consists of three sections.

The first section contains an analysis of network construction technologies for cottage complexes, an overview of sensor network standards, a study of existing solutions, and the formulation of technical requirements.

The second section is devoted to the development of the network scheme, configuration of its components, integration of IoT devices, and performance testing, including internet access, emergency notification, and sensor mechanisms.

The third section covers the configuration of the sensor network, implementation of video surveillance, emergency notification, NAT access to the Internet, and the network state prediction module; testing confirmed its functionality and readiness for use.

ЗМІСТ

ВСТУП

Сучасний розвиток інформаційних технологій значно підвищив вимоги до мережевої інфраструктури житлових комплексів, особливо у сфері автоматизації та забезпечення комфорту мешканців. Активне впровадження концепцій «розумного будинку» та «інтернету речей» (IoT) зумовлює потребу у створенні надійних і масштабованих мереж передавання даних, здатних обслуговувати велику кількість сенсорних пристроїв [1].

У котеджних комплексах, які охоплюють значні площі з численними будівлями, важливо забезпечити ефективне з'єднання між пристроями безпеки, енергетичного контролю, моніторингу та автоматизації побуту. Традиційні дротові мережі тут часто виявляються економічно невигідними й технічно складними, що стимулює використання бездротових технологій, зокрема сенсорних мереж.

Сенсорні мережі, побудовані на основі великої кількості малопотужних вузлів, забезпечують безперервний моніторинг, передачу даних та низьке енергоспоживання. Їхні ключові переваги – гнучкість та адаптація до змін середовища – роблять їх оптимальними для сучасних інфраструктурних рішень у житловому секторі [2].

Проєктування таких мереж є складним завданням, що вимагає врахування топології, протоколів зв'язку, надійності, безпеки та інтеграції з іншими системами [3]. Ефективність мережі визначається її здатністю обробляти велику кількість пристроїв та адаптуватися до змін.

Серед актуальних технологій реалізації бездротових сенсорних мереж – ZigBee, LoRaWAN, Z-Wave, Wi-Fi та NB-IoT. Вибір залежить від радіусу дії, потужності, пропускної здатності та рівня захисту даних.

Окремим викликом є створення програмного забезпечення для моніторингу, що дозволяє не лише відстежувати роботу пристроїв, а й виявляти відмови або зміни у структурі мережі. Такі системи мають бути простими у використанні, підтримувати віддалене адміністрування та забезпечувати масштабованість.

1 ТЕХНІЧНЕ ЗАВДАННЯ ТА ВИМОГИ ДО МЕРЕЖІ

1.1 Аналіз сучасних підходів до побудови мереж передачі даних для котеджних комплексів

Розвиток інформаційних технологій та зростаючий попит на автоматизацію житлових об’єктів спричинили значне ускладнення вимог до інфраструктури мереж передачі даних у котеджних комплексах. У сучасних підходах до побудови таких мереж необхідно враховувати не лише потреби користувачів у високошвидкісному доступі до Інтернету, але й вимоги до інтеграції сенсорних пристроїв, відеоспостереження, систем безпеки, енергоменеджменту та інших елементів «розумного будинку».

Мережева інфраструктура котеджного комплексу – це сукупність комунікаційних технологій, які забезпечують з’єднання всіх об’єктів комплексу в єдину інформаційну систему. Вона використовується для організації доступу до Інтернету, інтеграції сенсорних пристроїв, відеоспостереження, систем безпеки та автоматизації житлових будинків. Важливим аспектом є забезпечення стабільного бездротового з’єднання, яке дозволяє мінімізувати прокладання кабелів та спрощує розгортання нових пристроїв. Сенсорні мережі у таких комплексах відповідають за моніторинг навколишнього середовища, управління освітленням, кліматичними системами та безпекою мешканців. Оптимальне планування мережі дозволяє підвищити енергоефективність, зменшити експлуатаційні витрати та створити надійну систему управління. Проте розподілена топологія, необхідність підтримки високої якості обслуговування (QoS) та інтеграція бездротових рішень створюють низку технічних викликів.

На рисунку 1.1 наведено основні виклики мережевої інфраструктури котеджних комплексів, які необхідно враховувати під час проєктування. Одним із ключових аспектів є розподілена топологія, що передбачає з’єднання окремих котеджів у єдину мережеву систему та об’єднання обслуговуючих будівель у спільну інфраструктуру. Для ефективного функціонування мережі

використовуються технології сенсорних мереж, які забезпечують енергетичну ефективність та мінімізують споживання енергії.

Рисунок 1.1 – Основні виклики мережевої інфраструктури котеджних комплексів

Бездротовий доступ є невід'ємною складовою сучасних мереж у котеджних комплексах, оскільки дозволяє мінімізувати фізичне втручання у середовище, забезпечуючи покриття Wi-Fi та стабільний зв'язок для сенсорних пристроїв [4]. Важливим фактором є якість обслуговування, яка передбачає підтримку систем відеоспостереження, забезпечення гарантованого каналу зв'язку та ефективне функціонування систем безпеки. Усі ці виклики визначають специфіку побудови мереж у котеджних комплексах та впливають на вибір відповідних технологій.

Зважаючи на ці виклики, критично важливо обрати відповідну технологію побудови мережі, яка забезпечить стабільний зв'язок, достатню пропускну здатність та можливість інтеграції сенсорних пристроїв. Вибір архітектури комунікаційної інфраструктури безпосередньо впливає на якість обслуговування, ефективність передачі даних та гнучкість масштабування мережі. Саме тому необхідно розглянути сучасні технології побудови мереж передачі даних, що

визначають способи з’єднання будівель та обслуговування всієї цифрової екосистеми котеджного комплексу.

Технології побудови мереж передачі даних визначають спосіб з’єднання окремих будівель, ефективність передачі даних, стабільність роботи систем відеоспостереження, безпеки та сенсорних мереж [5]. У сучасних умовах використовується декілька основних підходів: дротові (оптоволоконні) мережі, бездротові технології (Wi-Fi, LTE, 5G) та гібридні рішення, які поєднують переваги обох підходів. Вибір тієї чи іншої технології залежить від масштабів комплексу, необхідної пропускної здатності та рівня безпеки.

При побудові мереж важливо враховувати як швидкість і стабільність підключення, так і вартість розгортання інфраструктури. Оптоволоконні лінії забезпечують високу пропускну здатність, проте їхнє прокладання може бути складним та дорогим. Бездротові технології, навпаки, простіші в розгортанні, але можуть мати проблеми з покриттям великих територій. Оптимальним підходом є використання гібридних рішень, які дозволяють поєднувати стабільність дротового зв’язку з мобільністю бездротових технологій.

На рисунку 1.2 наведено особливості технологій побудови мереж передачі даних. Основними напрямами є оптоволоконні, бездротові та гібридні рішення.

Рисунок 1.2 – Особливості технологій побудови мереж передачі даних

Оптоволоконні мережі (FTTH/FTTB) забезпечують високошвидкісне підключення, стабільний зв’язок та низькі затримки, однак їхнє впровадження вимагає значних фінансових витрат [6]. Бездротові технології, зокрема Wi-Fi (802.11ax), характеризуються високою пропускною здатністю, але можуть мати проблеми з покриттям великих територій. Для резервного з’єднання застосовуються LTE/5G-мережі. Найбільш ефективним підходом є гібридні рішення, що об'єднують дротові та бездротові технології, забезпечуючи стабільність підключення та можливість гнучкого розширення інфраструктури.

Оптоволоконні мережі є оптимальним вибором для магістрального з’єднання, а бездротові рішення забезпечують мобільність та швидке розгортання мережі. Гібридний підхід дозволяє поєднати переваги цих технологій, створюючи ефективну та надійну комунікаційну систему для котеджного комплексу.

Однак, для повноцінного функціонування мережевої інфраструктури котеджного комплексу недостатньо лише ефективного передавання даних між будівлями та користувачами. Важливим компонентом є впровадження сенсорних мереж, які забезпечують постійний моніторинг параметрів середовища, автоматизоване управління інженерними системами та інтеграцію елементів безпеки. Завдяки поєднанню сенсорних технологій із загальною мережею передавання даних створюється єдина інтелектуальна екосистема, що сприяє підвищенню комфорту мешканців та ефективному використанню ресурсів.

Сенсорні мережі є невід’ємною частиною сучасних котеджних комплексів, забезпечуючи автоматизований контроль та управління різними системами. Вони складаються з великої кількості автономних сенсорів, які збирають, аналізують і передають дані для підтримки безпеки, енергоефективності та комфорту мешканців. Такі мережі застосовуються для моніторингу мікроклімату в приміщеннях, контролю доступу, відеоспостереження, пожежної безпеки, розумного освітлення та інших важливих функцій. Завдяки можливості інтеграції з іншими мережевими технологіями сенсорні пристрої можуть працювати в складі єдиної цифрової екосистеми, забезпечуючи централізоване управління та аналітику. Використання сучасних бездротових протоколів, таких як ZigBee,

LoRaWAN, NB-IoT, дозволяє організувати ефективний зв’язок між пристроями при мінімальному енергоспоживанні. Оптимальне налаштування сенсорної мережі забезпечує стабільність передачі даних, захист інформації та можливість масштабування системи за потреби.

Розгортання сенсорних мереж у котеджних комплексах пов’язане з низкою технічних викликів, що потребують ретельного врахування при проєктуванні. На рисунку 1.3 представлені основні труднощі та перспективи розвитку сенсорних мереж у таких об’єктах. Однією з ключових проблем є забезпечення безперервного зв’язку та резервування каналів передачі даних для запобігання втратам інформації. Також важливою задачею є використання механізмів QoS для оптимізації мережевого трафіку, особливо для критичних додатків, таких як системи безпеки та відеоспостереження.

Рисунок 1.3 – Виклики та перспективи розвитку сенсорних мереж

Важливим аспектом функціонування сенсорних мереж є захист інформації, що передається між пристроями, оскільки бездротові з’єднання можуть бути вразливими до атак [7]. Для мінімізації ризиків несанкціонованого доступу

використовуються сучасні методи шифрування та аутентифікація пристроїв. Загальна стратегія безпеки сенсорних мереж передбачає багаторівневий підхід, що включає надійний контроль доступу, захист переданих даних та механізми виявлення загроз, забезпечуючи конфіденційність та стабільність функціонування системи.

Таким чином, використання сенсорних мереж у котеджних комплексах є необхідною умовою для ефективного функціонування розумних систем. Забезпечення безперервного зв’язку, оптимізація трафіку та високий рівень безпеки є ключовими факторами, які визначають ефективність та надійність таких мереж. Інноваційні технології та вдосконалені алгоритми управління дозволяють значно підвищити функціональність та гнучкість сенсорних мереж, роблячи їх основою інтелектуальної інфраструктури сучасного житлового середовища.

1.2 Стандарти сенсорних мереж та їх застосування при побудові моделі мережі

Після аналізу технічного завдання було визначено загальну структуру мережевої інфраструктури для котеджного комплексу. Основу функціональної схеми становить трирівнева архітектура з центральним хостовим вузлом і двома віддаленими сегментами – Містечко-1 та Містечко-2. У кожному з кластерів реалізовано локальну підмережу з підключенням сенсорних пристроїв до комутаторів через бездротову інфраструктуру. Усі вузли об’єднано через оптоволоконні канали, що забезпечують високошвидкісний обмін даними та резервування зв’язку.

У хостовій частині мережі розміщено головне серверне обладнання, включаючи маршрутизатор, центральний комутатор, сервер IoT, два WLC-контролери для управління бездротовим доступом, а також термінал оператора. У віддалених кластерних вузлах розміщено датчики руху, кліматичні пристрої,

камери відеоспостереження та кнопки аварійного виклику. Для кожного сегменту передбачено логічний поділ трафіку між користувачами й IoT через окремі VLAN. Підключення до глобальної мережі Інтернет організовано через зовнішній маршрутизатор з активованим NAT, що дозволяє безпечно транслювати локальні адреси.

Для підвищення надійності функціонування впроваджено альтернативний канал зв’язку між Містечком-1 та Містечком-2. Усі пристрої інтегруються в єдину інформаційну систему, яка централізовано керується сервером IoT. Структурна та функціональна побудова мережі зображена на рисунку 1.4. Після завершення побудови моделі виконано налаштування адресації, маршрутизації, логіки автоматичних дій, а також попереднє тестування для перевірки працездатності всієї архітектури в умовах сценарного навантаження.

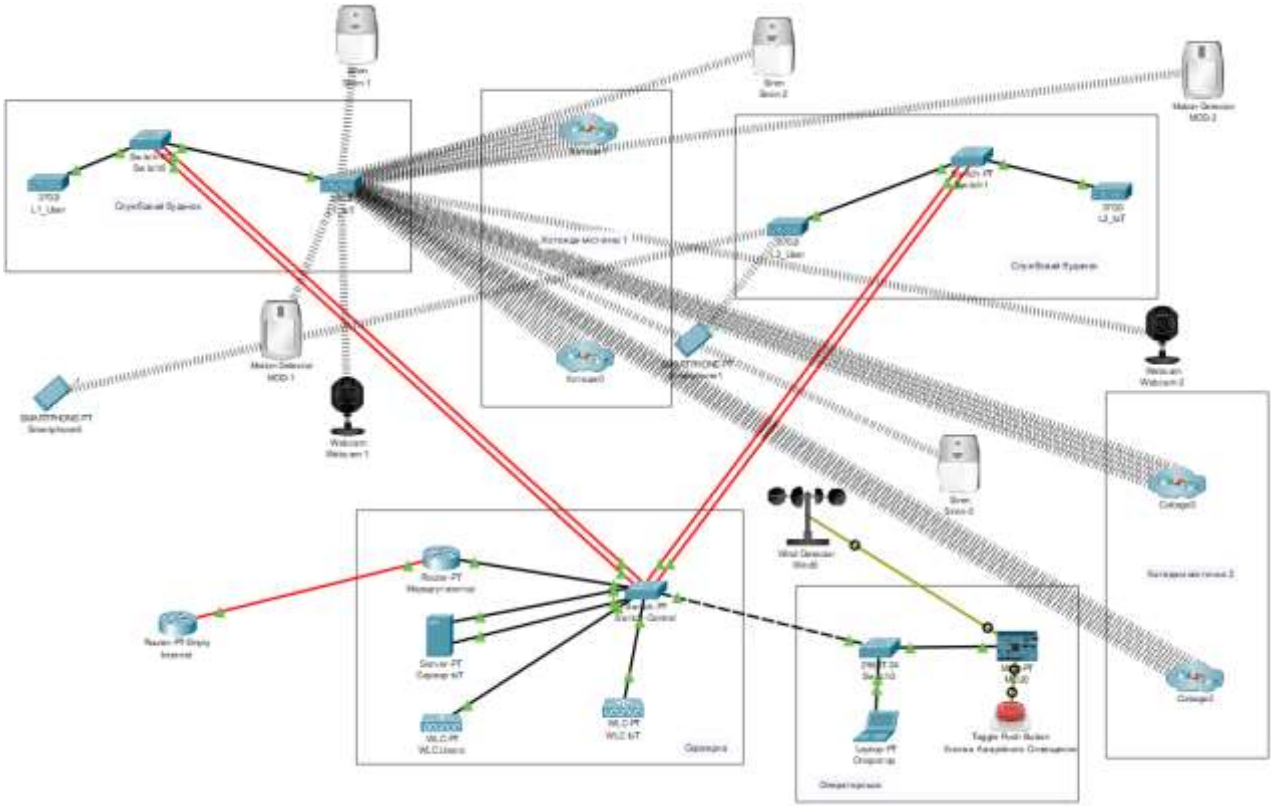


Рисунок 1.4 – Функціональна схема мережі

Для впровадження сенсорної інфраструктури в межах котеджного комплексу, окрім функціонального аспекту побудови, важливо також детально відобразити логічну структуру взаємозв’язків між ключовими компонентами

мережі. З цією метою побудовано структурну схему, яка ілюструє логічні зв’язки між усіма мережевими елементами, сервісними вузлами та сенсорними пристроями комплексу.

На рисунку 1.5 наведено структурну схему мережі, що відображає взаємозв’язки між центральним сервером, комутаторами, IoT-пристроями, маршрутизатором та користувацькими терміналами.

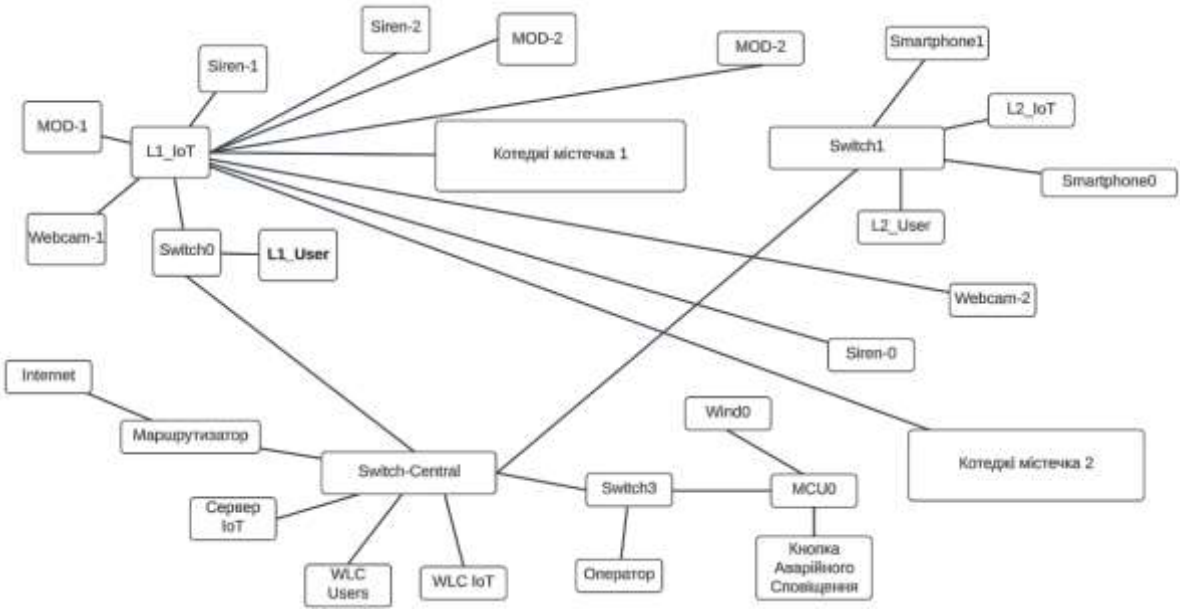


Рисунок 1.5 – Структурна схема мережі

Схема наочно демонструє поділ на підмережі IoT і User, централізоване управління через Switch-Central і наявність окремих кластерів для кожного з котеджних містечок.

1.3 Постановка завдань та визначення технічних вимог до мережі

Проєктування мережі передачі даних для котеджного комплексу передбачає створення надійної, масштабованої та безпечної інфраструктури, здатної підтримувати стабільне дротове та бездротове підключення користувачів і IoT-пристроїв. Мережа повинна забезпечувати централізоване управління, ефективний

розподіл ресурсів і можливість автоматизованого моніторингу та реагування на події. Основні технічні аспекти включають розробку фізичної та логічної архітектури, налаштування механізмів безпеки та моніторингу, а також інтеграцію систем аварійного сповіщення та сенсорних мереж.

Запропоноване рішення передбачає комплексний підхід до організації комунікаційної інфраструктури, який базується на використанні оптоволоконних каналів зв'язку для магістральних підключень, сегментованої бездротової мережі з VLAN-ізоляцією для користувачів та IoT-пристроїв, а також централізованого серверного хосту для керування та обробки даних. У межах цього проєкту необхідно розробити ефективну схему маршрутизації, налаштувати механізми безпеки, впровадити автоматизовану систему моніторингу, а також передбачити гнучкість розширення інфраструктури у разі збільшення кількості підключених пристроїв або розширення комплексу.

Вихідні технічні вимоги. Запропонована мережева архітектура повинна відповідати наступним вимогам:

– фізична інфраструктура:

Центральний сервер IoT, комутатори та маршрутизатори, а також WLC-контролери для керування точками доступу.

– мережеве розгалуження:

Два містечка-кластери, кожне з яких підключене до центрального хосту через оптоволоконні магістралі.

– бездротовий доступ:

Використання Wi-Fi 6 із поділом на VLAN-сегменти для окремих груп користувачів та IoT-систем.

– сенсорна мережа:

Підтримка підключення та управління інтелектуальними пристроями (камери спостереження, датчики руху, кліматичні системи).

– автоматизована система сповіщення:

Інтеграція з сенсорами для екстрених повідомлень та автоматичних сценаріїв реагування.

– захист інформації:

Використання механізмів NAT, VLAN-ізоляції, контролю доступу та шифрування даних.

– моніторинг та діагностика:

Централізоване ПЗ для відстеження продуктивності мережі, прогнозування несправностей та виведення рекомендацій щодо усунення проблем.

Функціональні вимоги. У межах проєкту передбачається:

Фізична організація мережі:

– оптоволоконні канали зв’язку між ключовими вузлами;

– гіперканали між центральним комутатором та комутаторами містечок.

Логічна архітектура:

– використання VLAN для розподілу трафіку;

– поділ мережі на сегменти для IoT-пристроїв та користувачів.

Бездротовий доступ:

– централізоване керування точками доступу через WLC-контролери;

– впровадження механізмів автентифікації користувачів.

Сенсорна інфраструктура:

– автоматичне підключення пристроїв IoT до серверного середовища;

– реагування на події (відеофіксація при русі, контроль мікроклімату).

Безпека мережі:

– VLAN-ізоляція для захисту критичних сегментів;

– використання NAT та списків контролю доступу (ACL).

Моніторинг та управління:

– автоматизований контроль продуктивності мережі;

– система прогнозування несправностей на основі машинного навчання.

Для підтримки стабільної роботи інфраструктури необхідно розробити систему аналізу та прогнозування мережевих станів. Ця система включатиме можливості збору та обробки даних у реальному часі, автоматичне навчання

моделей для виявлення потенційних збоїв, що дозволить запобігти аварійним ситуаціям. Основні функції моніторингової системи:

– навчання моделей машинного навчання для прогнозування несправностей;

– обробка відсутніх значень та створення аналітичних показників;

– збереження та управління версіями натренованих моделей;

– моніторинг критичних параметрів (температура, швидкість обертання, енергоспоживання);

– прогнозування стану вузлів мережі та формування звітів.

Результатом проєкту стане повноцінна мережева архітектура, що відповідає сучасним стандартам безпеки, продуктивності та масштабованості. Це рішення забезпечить:

– надійне підключення для мешканців та пристроїв IoT;

– контрольований доступ до мережевих ресурсів;

– гнучку можливість розширення інфраструктури у разі зростання кількості підключень;

– автоматизацію управління сенсорною мережею;

– вбудовані механізми безпеки та аварійного реагування.

Завдяки впровадженню цих рішень котеджний комплекс отримає ефективну, безпечну та масштабовану систему передачі даних, яка відповідатиме сучасним вимогам до інтелектуальних житлових середовищ.

2 ПРОЄКТУВАННЯ МЕРЕЖІ ПЕРЕДАЧІ ДАНИХ

2.1 Загальна схема котеджного комплексу з серверною частиною та двома містечками

Проєктування мережі передачі даних для котеджного комплексу передбачає створення надійної та масштабованої інфраструктури, що забезпечить стабільне

з’єднання між усіма підключеними пристроями. Архітектура мережі повинна підтримувати високу швидкість передавання даних, ефективний розподіл навантаження та централізоване управління всіма IoT-пристроями. Важливим аспектом є використання оптоволоконних магістралей, що забезпечують високу пропускну здатність та мінімізують затримки при передаванні даних між центральним серверним хостом та окремими містечками. Дана архітектура дозволяє інтегрувати сенсорні мережі, системи безпеки та автоматизовані рішення у єдину інформаційну систему, що робить її ідеальною для сучасного розумного котеджного комплексу.

Для досягнення цих цілей необхідно розробити ефективну топологію мережі, яка забезпечить взаємодію всіх компонентів системи, оптимальну маршрутизацію трафіку та резервування каналів зв’язку. Відповідна структурна схема мережевої інфраструктури котеджного комплексу представлена на рисунку 2.1.

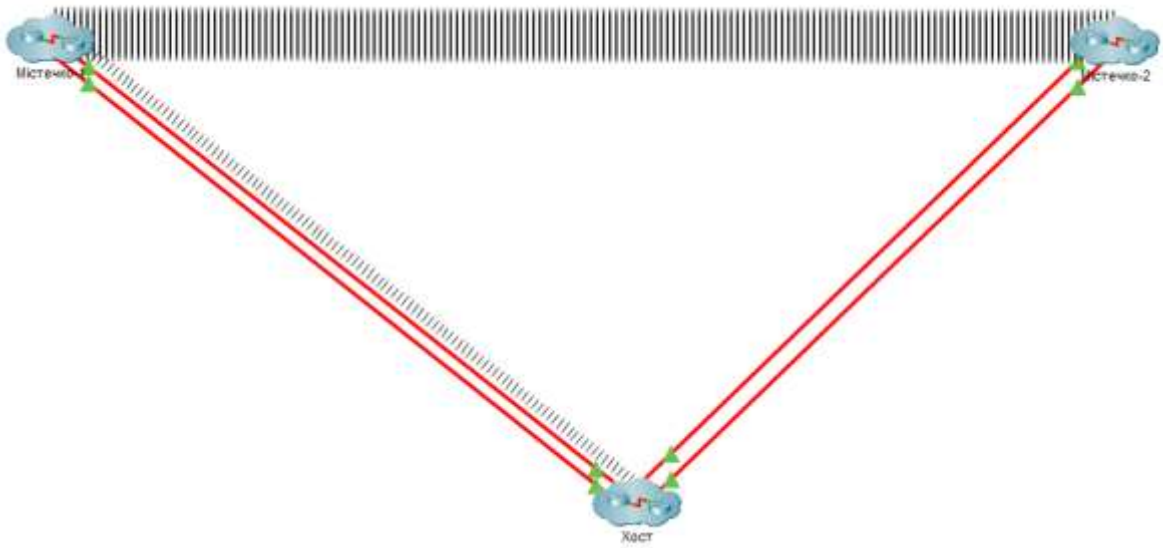


Рисунок 2.1 – Загальна схема мережі

На схемі зображено трирівневу архітектуру, що охоплює центральний кластер Хост і два віддалені вузли – Містечко-1 та Містечко-2. З’єднання між ними здійснюється через оптоволоконні канали, що забезпечують високошвидкісний обмін даними та резервування. У центрі архітектури розміщено серверну частину з

комутатором, маршрутизатором, сервером IoT та контролерами WLC для керування Wi-Fi.

Містечка включають житлові будинки з розумними кімнатами, де встановлено IoT-пристрої: системи освітлення, клімат-контролю, відеоспостереження та аварійного сповіщення. Всі пристрої підключаються через окремі VLAN-сегменти, а керування здійснюється централізовано через IoT-сервер на Хості.

Схема також передбачає альтернативний канал між містечками, що підвищує відмовостійкість мережі. Така топологія забезпечує стабільність, можливість масштабування та інтеграцію нових IoT-рішень без втрати якості обслуговування користувачів.

2.1.1 Схема серверної

Центральним вузлом мережевої інфраструктури котеджного комплексу є серверна, що розташована на Хості. Це приміщення містить усі необхідні мережеві пристрої для управління передачею даних, маршрутизації трафіку, бездротового доступу та інтеграції IoT-пристроїв. Основною задачею серверної є забезпечення стабільного функціонування всієї комунікаційної системи, балансування навантаження між підключеними сегментами мережі та надання доступу до Інтернету. Важливими компонентами інфраструктури серверної є центральний мережевий комутатор, головний маршрутизатор, сервер IoT, а також контролери бездротового зв’язку.

Для забезпечення повноцінного функціонування серверної розроблена її структурна схема, наведена на рисунку 2.2.

На схемі що зазначена нижче представлено основні компоненти серверної, операторської та їх взаємозв’язки з іншими сегментами мережі. Головним елементом є центральний комутатор (Switch-Central), який виконує функції розподілу мережевого трафіку між містечками та серверними пристроями. До комутатора підключені два оптоволоконні гіперканали, які забезпечують високошвидкісне з’єднання між серверною та віддаленими містечками.

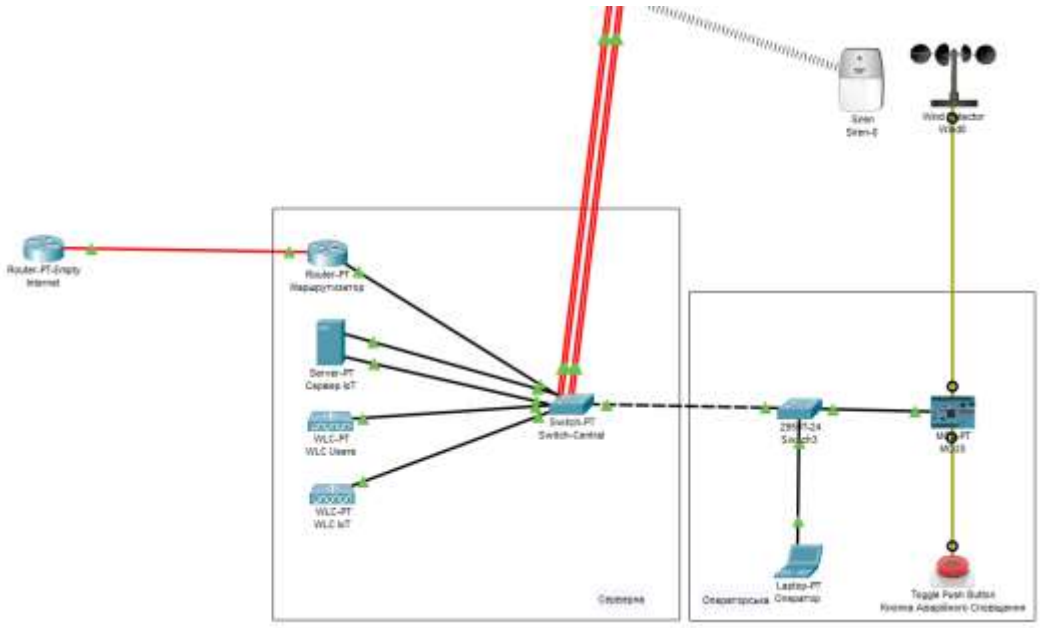


Рисунок 2.2 – Схема хосту

До центрального комутатора підключений головний маршрутизатор (Router-PT), який здійснює маршрутизацію трафіку між внутрішніми сегментами мережі та зовнішніми ресурсами. Він також виконує функції DHCP-сервера, що розподіляє IP-адреси для всіх клієнтів мережі, забезпечуючи коректну маршрутизацію трафіку.

Серверна також містить сервер IoT (Server-PT), який відповідає за обробку даних від розумних пристроїв, керування автоматизованими системами та централізоване управління мережею сенсорів. Для організації бездротового доступу до мережі використовується два контролери WLC (Wireless LAN Controller) – один для звичайних клієнтів мережі, другий для підключення IoT-пристроїв. Це рішення забезпечує розподіл бездротового трафіку та підвищує рівень безпеки, ізолюючи сенсорні мережі від загального користувацького сегмента.

У операторській, яка є частиною серверного вузла, встановлено додатковий комутатор, що забезпечує підключення оператора системи та IoT-пристроїв, які використовують дротовий спосіб комунікації. До нього підключений робочий ноутбук оператора (Laptop-PT), що дозволяє здійснювати моніторинг мережі та адміністрування обладнання в реальному часі. Також у цьому сегменті мережі

розташовані сенсорні пристрої аварійного сповіщення – метеостанція (Wind Sensor), сирена (Siren-0) та кнопка екстреного виклику (Toggle Push Button).

Серверна підключена до Інтернету через головний маршрутизатор, що дозволяє забезпечувати зв’язок як для користувачів, так і для IoT-пристроїв, при цьому використовується мережевий NAT для безпечного доступу до глобальної мережі. Всі взаємозв’язки між пристроями організовані таким чином, щоб забезпечити резервування каналів зв’язку та мінімізувати вплив можливих відмов окремих сегментів мережі.

2.1.2 Схема містечка

Для забезпечення стабільного підключення мешканців та IoT-пристроїв у межах котеджного комплексу передбачено розгортання окремої мережевої інфраструктури для кожного містечка. Основним завданням цієї частини мережі є організація бездротового доступу для користувачів і сенсорних пристроїв, інтеграція відеоспостереження, а також підтримка систем безпеки. Завдяки використанню оптоволоконних магістралей і централізованих точок управління досягається баланс між продуктивністю та гнучкістю управління мережею.

Функціональна схема мережі містечка представлена на рисунку 2.3.

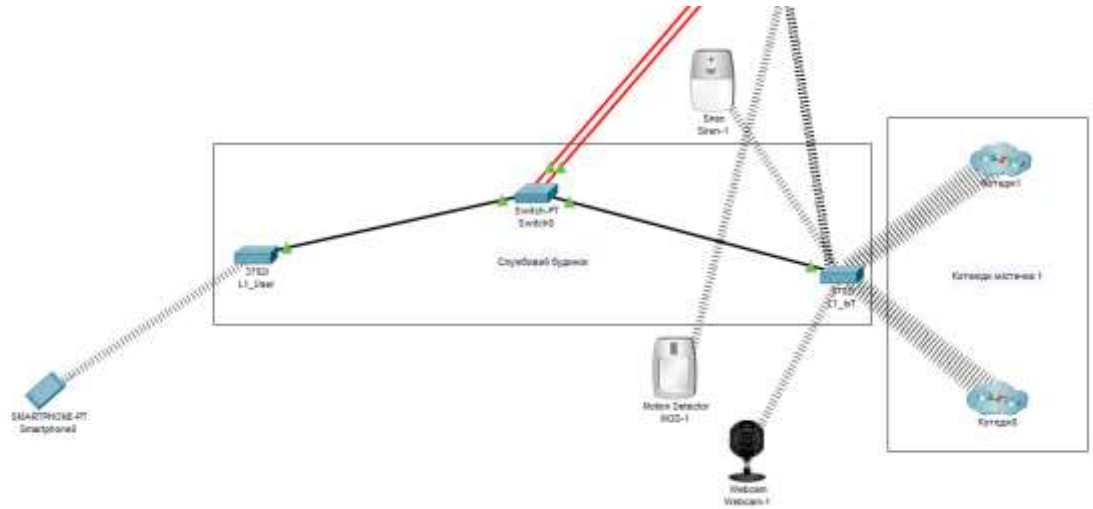


Рисунок 2.3 – Схема Містечка-1

На схемі зображена топологія підключення містечка до основного комунікаційного вузла. Центральним елементом є мережевий комутатор (Switch-

3), який формує гіперканал, побудований із двох оптичних каналів зв’язку, що підключаються до центрального комутатора хосту. Це забезпечує високу пропускну здатність та мінімізує затримки в обміні даними.

До комутатора підключені дві бездротові точки доступу: L1_User, яка використовується для підключення звичайних клієнтів мережі, та L1_IoT, призначена для IoT-пристроїв. Такий підхід дозволяє ізолювати трафік сенсорної мережі від основного користувацького сегмента, що підвищує рівень безпеки та запобігає перевантаженню мережі.

У межах містечка до бездротової інфраструктури підключені кластерні вузли котеджів Котедж-0 та Котедж-1, які взаємодіють із відповідними точками доступу. Завдяки цьому забезпечується безперебійний зв’язок для розумних будинкових систем, таких як освітлення, клімат-контроль, система доступу та відеоспостереження.

Окрім основних точок доступу, у схемі передбачено окремі IoT-пристрої, що виконують функції безпеки та моніторингу. До них відносяться сирена (Siren-0), яка активується у разі виявлення загрози, детектор руху (Motion Detector-1), що виявляє активність у зоні покриття, та відеокамера (Webcam-1), яка здійснює запис та трансляцію відео. Ці пристрої інтегровані в єдину мережу та передають дані до центрального сервера IoT для обробки та реагування.

Завдяки такій архітектурі забезпечується стабільний зв’язок, контроль безпеки та централізоване управління IoT-пристроями. Використання окремих VLAN для різних категорій пристроїв дозволяє гнучко керувати трафіком та запобігати можливим мережевим конфліктам. Запропоноване рішення також дозволяє легко масштабувати мережу містечка, додаючи нові котеджі, точки доступу або сенсорні пристрої без значних змін у загальній архітектурі системи.

2.1.3 Схема котеджу

Для керування мікрокліматом та безпекою в кожному котеджі котеджного комплексу передбачено використання IoT-пристроїв, які автоматизують основні процеси. Всі пристрої взаємодіють між собою через бездротову мережу та

централізовано керуються з головного сервера IoT, що забезпечує автономну роботу системи та можливість дистанційного управління.

Функціональна схема розумної кімнати представлена на рисунку 2.4.

На схемі зображені основні IoT-пристрої, що використовуються в розумній кімнаті котеджу. Всі вони підключені до центрального сервера управління, що дозволяє автоматично коригувати роботу систем відповідно до умов навколишнього середовища та налаштувань користувача.

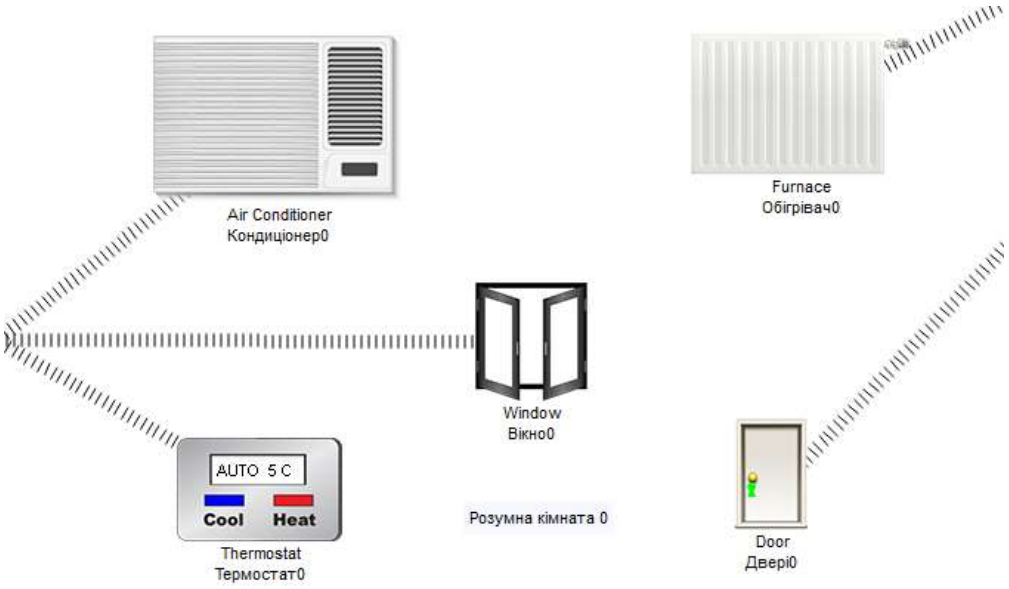


Рисунок 2.4 – Схема Котеджу-0

Ключовим елементом системи є термостат (Thermostat0), який регулює температуру в приміщенні та здійснює автоматичний вибір режиму нагрівання або охолодження. Він взаємодіє з кондиціонером (Air Conditioner0), що відповідає за підтримку комфортного мікроклімату в кімнаті, та обігрівачем (Furnace0), який використовується в холодний період для забезпечення оптимальної температури.

Також у схемі передбачено вікно (Window0) та двері (Door0), які можуть бути інтегровані в систему безпеки та енергоефективності. Наприклад, якщо вікно відкрите, термостат може автоматично вимкнути обігрівач або кондиціонер для запобігання марнотратного використання енергії. Водночас дверний сенсор може використовуватися для автоматичного ввімкнення освітлення або активації сигналізації у разі несанкціонованого доступу.

Усі пристрої пов’язані між собою та працюють у єдиній системі розумного дому. Керування ними здійснюється через бездротовий зв’язок із сервером IoT, що дозволяє аналізувати поточні параметри, прогнозувати зміни та оптимізувати роботу всіх компонентів. Така інтегрована система значно підвищує рівень комфорту мешканців та сприяє ефективному використанню ресурсів у котеджному комплексі.

2.2 Планування безшовного бездротового доступу для сенсорної та користувацької мережі з розділенням на рівні VLAN

У мережевій архітектурі передбачено два основні типи бездротового доступу: для звичайних користувачів та для IoT-пристроїв, що дозволяє забезпечити розподіл трафіку та підвищити рівень інформаційної безпеки. Для реалізації логічного поділу цих категорій клієнтів у мережі впроваджено сегментацію за допомогою віртуальних локальних мереж (VLAN), що дозволяє оптимізувати маршрутизацію трафіку, запобігати перевантаженню каналів зв’язку та забезпечити ізоляцію критично важливих IoT-пристроїв від загального користувацького сегмента. VLAN-сегментація реалізується на рівні трьох мережевих комутаторів, де кожному типу трафіку присвоєно окремий VLAN з відповідною адресацією, що наведено у таблиці 2.1.

Таблиця 2.1 – Список VLAN мережі

VLAN	Тип	Адресація
100	Користувачі	192.168.0.0/24
101	IoT-прилади	192.168.1.0/24

Для забезпечення трафік-менеджменту оптоволоконні гіперканали, що з’єднують мережеві комутатори містечок із центральним комутатором, а також магістральний канал між центральним комутатором та головним

маршрутизатором, буде переведено у режим транку. Це дозволить передавати трафік усіх VLAN через магістральні канали зв’язку відповідно до налаштувань, зазначених у таблиці 2.1.

Інші порти комутаторів будуть переведені у режим доступу (Access Mode), що забезпечить підключення клієнтських пристроїв до відповідних VLAN згідно з їхнім призначенням. Окрему увагу приділено серверній інфраструктурі: основний сервер буде підключений до центрального комутатора двома каналами, оскільки на ньому розміщений DNS-сервер, що обслуговує обидва VLAN. Такий підхід дозволить підвищити відмовостійкість системи та забезпечити швидкий і стабільний доступ до критичних сервісів для всіх учасників мережі.

2.3 Налаштування мережі

2.3.1 Розробка схеми адресації

Відповідно до визначеної VLAN-сегментації (табл. 2.2), необхідно розробити детальну схему адресації для ключових мережевих ресурсів у кожному з логічних сегментів. Такий підхід забезпечує чітку організацію мережевого трафіку, ефективне керування підключеними пристроями та оптимізує процес маршрутизації. Впровадження автоматичного розподілу IP-адрес через DHCP-сервер, централізованого керування точками бездротового доступу через WLC-контролери та сегментованого обслуговування користувачів і IoT-пристроїв дозволяє підвищити продуктивність мережі та забезпечити її стабільне функціонування. У таблиці 2.2 наведено схему адресації основних ресурсів VLAN.

Таблиця 2.2 – Схема адресації ресурсів VLAN

VLAN	DHCP	DNS	WLC	AP Містечко-1	AP Містечко-2
100	Маршрутизатор 192.168.0.1	Сервер IoT 192.168.0.2	WLC Users 192.168.0.4	L1_Users 192.168.0.10	L2_Users 192.168.0.11
101	Маршрутизатор 192.168.1.1	Сервер IoT 192.168.1.2	WLC IoT 192.168.1.4	L1_IoT 192.168.1.10	L2_IoT 192.168.1.11

Згідно з представленою схемою, роль DHCP-сервера виконує головний маршрутизатор, який автоматично розподіляє IP-адреси серед підключених пристроїв у межах кожного VLAN. DNS-сервер розташований на сервері IoT, що забезпечує розв’язування доменних імен та взаємодію мережевих пристроїв між собою.

Керування бездротовою мережею здійснюється через контролери WLC: один для користувачів (WLC Users, 192.168.0.4), другий для IoT-пристроїв (WLC IoT, 192.168.1.4). Така сегментація дозволяє розділити трафік і мінімізувати ризики впливу користувацького трафіку на критично важливі IoT-пристрої.

В обох містечках передбачено окремі точки бездротового доступу (AP) для звичайних користувачів та IoT-пристроїв. У Містечку-1 точки доступу для користувачів (L1_Users) та IoT-пристроїв (L1_IoT) отримують адреси 192.168.0.10 та 192.168.1.10 відповідно. Аналогічно, у Містечку-2 точки L2_Users та L2_IoT отримують адреси 192.168.0.11 та 192.168.1.11.

Завдяки такому підходу мережа забезпечує логічну ізоляцію трафіку, підвищує рівень безпеки та дозволяє централізовано управляти всіма підключеними пристроями через визначені адресні простори VLAN.

2.3.2 Налаштування центрального комутатора та комутаторів містечок

Для забезпечення ефективного функціонування мережі необхідно налаштувати центральний комутатор, який виконує ключову роль у маршрутизації та сегментації трафіку між користувачами та IoT-пристроями. Налаштування комутатора передбачає створення гіперканалів для магістрального з’єднання, визначення VLAN-сегментів та конфігурацію режимів портів для оптимального управління трафіком.

Одним із важливих етапів є створення агрегованих каналів (Port-channel), які забезпечують високу пропускну здатність та підвищену відмовостійкість з'єднань між центральним комутатором та комутаторами містечок. Гіперканали формуються через групування окремих фізичних портів у логічні канали, що дозволяє балансувати навантаження та мінімізувати ймовірність перевантажень. Усі магістральні з’єднання функціонують у режимі транку, що дозволяє передавати

трафік різних VLAN через один фізичний канал; відповідні налаштування інтерфейсів центрального комутатора представлені у додатку А.

Додатково здійснюється конфігурація окремих портів для підключення пристроїв відповідно до їхньої VLAN-приналежності. Усі інтерфейси магістрального підключення налаштовані в режимі trunk, що дозволяє передавати трафік VLAN 100 (користувачі) та VLAN 101 (IoT-пристрої). Порти, які безпосередньо підключаються до клієнтських пристроїв або серверного обладнання, налаштовані в режимі access та закріплені за відповідним VLAN, що наведено у лістингу 2.1.

Лістинг 2.1 – Налаштування комутатора Містечка-1/Містечка-2

```
interface Port-channel1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
 channel-group 1 mode desirable
!
interface GigabitEthernet1/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
 channel-group 1 mode desirable
!
interface GigabitEthernet2/1
 switchport access vlan 100
 switchport mode access
!
interface GigabitEthernet3/1
 switchport access vlan 101
 switchport mode access
!
…
!
interface Vlan100
 no ip address
!
interface Vlan101
 no ip address
```

Окремо створено віртуальні інтерфейси VLAN100 та VLAN101, що використовуються для логічного поділу трафіку, забезпечення ізоляції між

сегментами та організації централізованого керування підключеними пристроями. Наявність VLAN-ізоляції дозволяє підвищити рівень безпеки та запобігти несанкціонованому доступу до критично важливих компонентів мережі.

Завдяки такій конфігурації центральний комутатор ефективно розподіляє навантаження між сегментами мережі, забезпечує стабільну маршрутизацію даних та підтримує масштабованість мережевої інфраструктури. Використання агрегованих каналів та VLAN-сегментації дозволяє забезпечити оптимальну продуктивність мережі та гнучкість її налаштування під подальше розширення котеджного комплексу.

2.3.3 Налаштування WLC та точок доступу сенсорної та користувацької мережі

Для забезпечення стабільного бездротового з’єднання у котеджному комплексі використовуються WLC, які відповідають за централізоване керування точками доступу, балансування навантаження та безпеку підключень. Оскільки у проєктованій мережі передбачено два окремі VLAN для користувачів та IoT-пристроїв, використовується два WLC-контролери: один для клієнтських пристроїв, другий – для сенсорної мережі. Такий підхід дозволяє мінімізувати затримки у роботі бездротової інфраструктури та розділити трафік між двома категоріями підключених пристроїв.

Для коректної роботи бездротової мережі виконується налаштування WLC-контролерів, що включає конфігурацію IP-адрес, шлюзів за замовчуванням, підключення до DNS-сервера та визначення політики доступу для точок бездротового підключення. Конфігурація одного з таких контролерів, призначеного для користувацької мережі (WLC Users), представлена у лістингу 2.2.

Лістинг 2.2 – Налаштування мережі управління WLC Users

```
config interface address management 192.168.0.4 255.255.255.0
192.168.0.1
config network dns primary 192.168.0.2
config network dns secondary <IP_address_of_secondary_DNS>
config hostname YourWLCName
config interface virtual 1.1.1.1
save config
```

```
reset system
```

Основним параметром конфігурації є IPv4-адреса контролера, яка у цій мережі має значення 192.168.0.4, що відповідає схемі адресації ресурсів VLAN. Для правильної маршрутизації трафіку встановлено маску підмережі 255.255.255.0, що визначає доступність пристроїв у межах підмережі користувачів. Шлюзом за замовчуванням призначено 192.168.0.1, тобто головний маршрутизатор, який забезпечує вихід у глобальну мережу.

Контролер також налаштований на використання DNS-сервера, IP-адреса якого 192.168.0.2, що відповідає розташованому у мережі серверу IoT. Це дозволяє точкам доступу та підключеним пристроям правильно визначати адреси внутрішніх мережевих ресурсів.

У лістингу. 2.3 представлена конфігурація WLC IoT, який забезпечує централізоване керування точками доступу, що підключаються до VLAN 101 для IoT-пристроїв. Контролер має IP-адресу 192.168.1.4, що відповідає схемі адресації IoT-сегмента. Маска підмережі встановлена як 255.255.255.0, а шлюз за замовчуванням – 192.168.1.1, що відповідає головному маршрутизатору.

Лістинг 2.3 – Налаштування мережі управління WLC IoT

```
configure terminal
interface vlan 1
ip address 192.168.1.4 255.255.255.0
exit
ip default-gateway 192.168.1.1
ip name-server 192.168.1.2
exit
write memory
```

В лістингу 2.4 представлено налаштування бездротової точки доступу L1_User, яка підключається до VLAN 100 для користувачів. Відповідно до таблиці 2.2, точці призначено статичну IP-адресу 192.168.0.10 з маскою підмережі 255.255.255.0.

Також налаштовано основний контролера WLC Users для точки доступу L1_User. Як Primary Controller вказано 192.168.0.4, що відповідає IP-адресі контролера для користувацької бездротової мережі згідно з таблицею 2.2.

Лістинг 2.4 – Налаштування адреси згідно таблицею 2.2 для L1_User

```
enable
configure terminal

interface GigabitEthernet0
 description L1_User - MAC 0050.0F52.0801
no shutdown
 ip address 192.168.0.10 255.255.255.0
 duplex auto
 speed auto
exit

wireless ap name L1_User
 wireless management interface vlan 100
 capwap ap primary-base WLC_Users 192.168.0.4

end
write memory
```

Порт підключення GigabitEthernet0 активовано, встановлено автоматичне визначення швидкості та режиму дуплексу, що забезпечує оптимальну продуктивність. Завдяки такій конфігурації точка доступу працює у визначеному сегменті мережі та ефективно взаємодіє з контролером WLC Users, забезпечуючи бездротовий доступ для мешканців котеджного комплексу.

Ця конфігурація забезпечує централізоване управління точкою доступу через WLC, що дозволяє автоматично оновлювати налаштування, балансувати навантаження та підтримувати стабільний бездротовий зв’язок у містечку.

Рисунку 2.5 відображає створення нової групи точок доступу на WLC Users. Група отримала назву AP_User, а її точки доступу L1_User та L2_User успішно додані та мають статус Online, що свідчить про коректне з'єднання з контролером.

У цій конфігурації для групи точок доступу встановлено SSID «AP_User», який буде використовуватися для автентифікації користувачів. Це дозволяє

централізовано керувати всіма точками доступу у межах користувацької мережі, оптимізуючи їхню продуктивність і покриття.

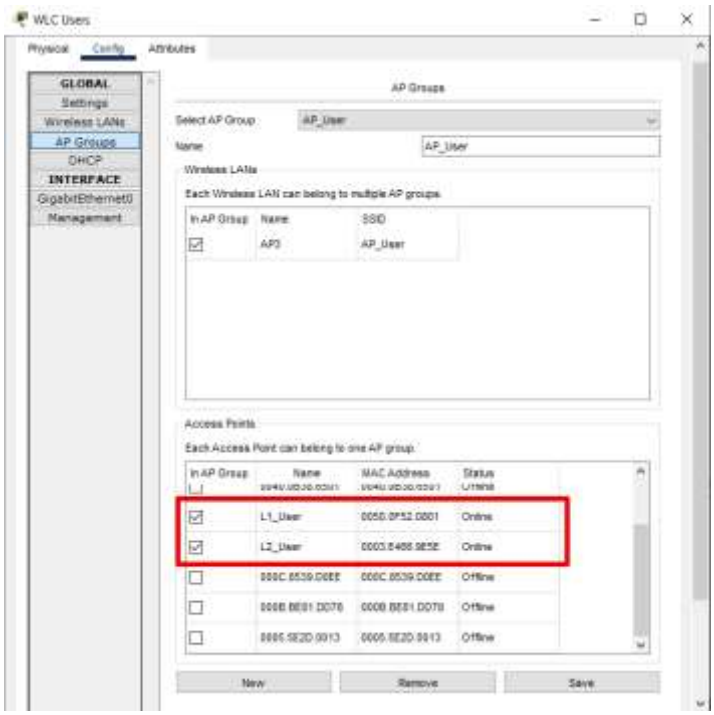


Рисунок 2.5 – Створення нової групи точок доступу

На лістингу 2.5 наведено налаштування SSID та параметрів автентифікації для бездротової мережі користувачів на WLC Users. SSID встановлено як «AP_User», що відповідає створеній групі точок доступу.

Лістинг 2.5 – Налаштування ім’я SSID та параметри автентифікації

```
configure terminal
wlan AP_User 1 AP_User
vlan 100
security wpa akm psk set-key ascii 0 12345678
security wpa akm psk enable
security wpa cipher aes enable
no shutdown
exit
ap name L1_User
ap group-name AP_User
interface GigabitEthernet0
ip address 192.168.0.10 255.255.255.0
no shutdown
exit
```

Для забезпечення безпеки підключень використовується протокол WPA2-PSK, який передбачає автентифікацію користувачів за допомогою парольного захисту. Вибрано шифрування AES, що забезпечує високий рівень захисту даних у бездротовій мережі. Така конфігурація дозволяє гарантувати безпечний доступ користувачів до мережі котеджного комплексу та централізовано керувати параметрами бездротового з’єднання.

2.3.4 Налаштування інтерфейсів та DHCP-серверів маршрутизатора

Для забезпечення коректної роботи мережі необхідно налаштувати інтерфейси маршрутизатора та організувати автоматичний розподіл IP-адрес за допомогою DHCP-серверів для кожного VLAN. Оскільки у проєктованій мережі передбачено два окремі VLAN-сегменти – для користувачів (VLAN 100) та IoT-пристроїв (VLAN 101), маршрутизатор має відповідати за правильне розмежування трафіку, підтримку міжмережевої маршрутизації та надання IP-адрес клієнтам.

Кожен із логічних сегментів отримує окрему підмережу, а зв'язок між ними забезпечується через підінтерфейси маршрутизатора, налаштовані для роботи у режимі 802.1Q. Така конфігурація дозволяє маршрутизатору приймати трафік із тегами VLAN та направляти його у відповідний сегмент мережі без зміни фізичної топології. Детальне налаштування інтерфейсів маршрутизатора наведено у лістингу 2.6.

Лістинг 2.6 – Налаштування інтерфейсів маршрутизатора

```
interface GigabitEthernet1/0
 no ip address
 duplex auto
 speed auto
!
interface GigabitEthernet1/0.100
 encapsulation dot1Q 100
 ip address 192.168.0.1 255.255.255.0
!
interface GigabitEthernet1/0.101
 encapsulation dot1Q 101
 ip address 192.168.1.1 255.255.255.0
```

Фізичний інтерфейс GigabitEthernet1/0 налаштований без IP-адреси та використовується для створення логічних підінтерфейсів, що здійснюють маршрутизацію між VLAN-сегментами.

Підінтерфейс GigabitEthernet1/0.100 працює у режимі 802.1Q для VLAN 100, отримує тег 100 та призначений для обслуговування користувацької мережі. Йому присвоєно IP-адресу 192.168.0.1/24, яка виконує функцію шлюзу за замовчуванням для клієнтів цього сегмента.

Аналогічно, підінтерфейс GigabitEthernet1/0.101 працює для VLAN 101, має тег 101 та IP-адресу 192.168.1.1/24, яка є шлюзом для IoT-пристроїв.

Для автоматичної конфігурації IP-адрес клієнтів у кожному VLAN використовується вбудований DHCP-сервер маршрутизатора, що дозволяє централізовано призначати IP-адреси підключеним пристроям. Налаштування DHCP-сервера забезпечує спрощене управління мережею, знижує ризик конфліктів адрес та мінімізує необхідність ручного налаштування кожного клієнта.

У проєктованій мережі передбачено два окремі DHCP-пули:

– net0 для користувацького VLAN (192.168.0.0/24);
– net1 для IoT-пристроїв у VLAN (192.168.1.0/24).

Для кожного DHCP-пулу задається шлюз за замовчуванням (default-router), який відповідає за маршрутизацію трафіку у відповідному сегменті, а також вказується DNS-сервер, що забезпечує коректне розв’язування доменних імен.

Щоб уникнути конфліктів з адресами критичних мережевих пристроїв (маршрутизатор, сервери, контролери), перші 20 IP-адрес у кожному сегменті зарезервовані та не видаються клієнтам. Конфігурація DHCP-серверів маршрутизатора приведена у лістингу 2.7.

Лістинг 2.7 – Налаштування DHCP-серверів маршрутизатора

```
ip dhcp excluded-address 192.168.1.1 192.168.1.20
ip dhcp excluded-address 192.168.0.1 192.168.0.20
!
ip dhcp pool net0
 network 192.168.0.0 255.255.255.0
 default-router 192.168.0.1
 dns-server 192.168.0.2
```

```
ip dhcp pool net1
 network 192.168.1.0 255.255.255.0
 default-router 192.168.1.1
 dns-server 192.168.1.2
!
```

На початку виконується виключення зарезервованих IP-адрес, щоб вони не призначалися динамічно. У сегменті 192.168.0.0/24 резервуються адреси 192.168.0.1 – 192.168.0.20, а у сегменті 192.168.1.0/24 – 192.168.1.1 – 192.168.1.20. Далі створюються DHCP-пули для обох VLAN. У пулі net0 задається адресний простір 192.168.0.0/24, маршрутизатор 192.168.0.1 як шлюз за замовчуванням, а також DNS-сервер 192.168.0.2, що відповідає за обслуговування імен у цьому сегменті. Аналогічно, для net1 використовується підмережа 192.168.1.0/24, шлюз 192.168.1.1 та DNS-сервер 192.168.1.2.

Ця конфігурація забезпечує гнучке управління IP-адресами, дозволяє зменшити навантаження на адміністраторів мережі та гарантує стабільне підключення всіх клієнтів у котеджному комплексі. DHCP-сервер автоматично розподіляє адреси серед користувачів і IoT-пристроїв, підтримуючи логічне розділення трафіку та підвищуючи загальну продуктивність мережі.

2.3.5 Налаштування кінцевого обладнання

Конфігурація включає статичну та динамічну маршрутизацію, налаштування параметрів безпеки та підключення до відповідних VLAN-сегментів. Впровадження цих налаштувань гарантує оптимальну роботу сенсорних пристроїв, мобільного обладнання та інших клієнтів мережі.

Для коректної роботи IoT-пристроїв необхідно виконати налаштування бездротового з'єднання, призначити відповідні параметри аутентифікації та шифрування, а також забезпечити правильний розподіл IP-адрес. Конфігурація кінцевого обладнання зображена на лістингу. 2.8.

Лістинг 2.8 –Налаштування приладу IoT на AP_Iot

```
interface Wireless0
ssid AP_lot
encryption mode ciphers aes
authentication open
```

```
authentication key-management wpa version 2
wpa-psk ascii 0 12345678
!
bandwidth 300000
mac-address 000D.5852.7991
no shutdown
!
ip address dhcp
!
ipv6 address FE80::2D0:58FF:FE52:7991 link-local
!
```

У лістингу розписані налаштування бездротового IoT-пристрою MOD-1, який підключається до мережі AP_IoT. SSID встановлено як «AP_IoT», що відповідає бездротовій мережі для сенсорних пристроїв. Для автентифікації використовується WPA2-PSK із паролем «12345678», а для шифрування даних застосовується AES, що забезпечує високий рівень захисту підключення.

Пристрій підтримує швидкість з'єднання до 300 Mbps, що є достатнім для стабільної передачі даних у межах IoT-сегмента мережі. IP-адреса 192.168.1.48 була призначена вручну, оскільки для деяких пристроїв IoT доцільно використовувати статичну конфігурацію, що дозволяє уникати конфліктів адрес та спрощує управління мережею. Маска підмережі встановлена 255.255.255.0, що відповідає адресному простору VLAN 101.

Додатково у конфігурації передбачено підтримку IPv6-адресації, де пристрій отримав локальну адресу FE80::2D0:58FF:FE52:7991. Використання IPv6 сприяє підвищенню гнучкості мережевої інфраструктури та розширенню можливостей для майбутніх інтеграцій з іншими пристроями.

На рисунку 2.6 представлено налаштування робочої машини (PM) оператора, яка підключається до мережі через дротовий інтерфейс FastEthernet0.

Для коректної роботи мережі встановлено шлюз за замовчуванням 192.168.1.1, що відповідає маршрутизатору у VLAN 101 (мережа IoT-пристроїв та операторських станцій). Також вказано DNS-сервер 192.168.1.2, що використовується для розв’язування доменних імен та взаємодії з іншими пристроями мережі.

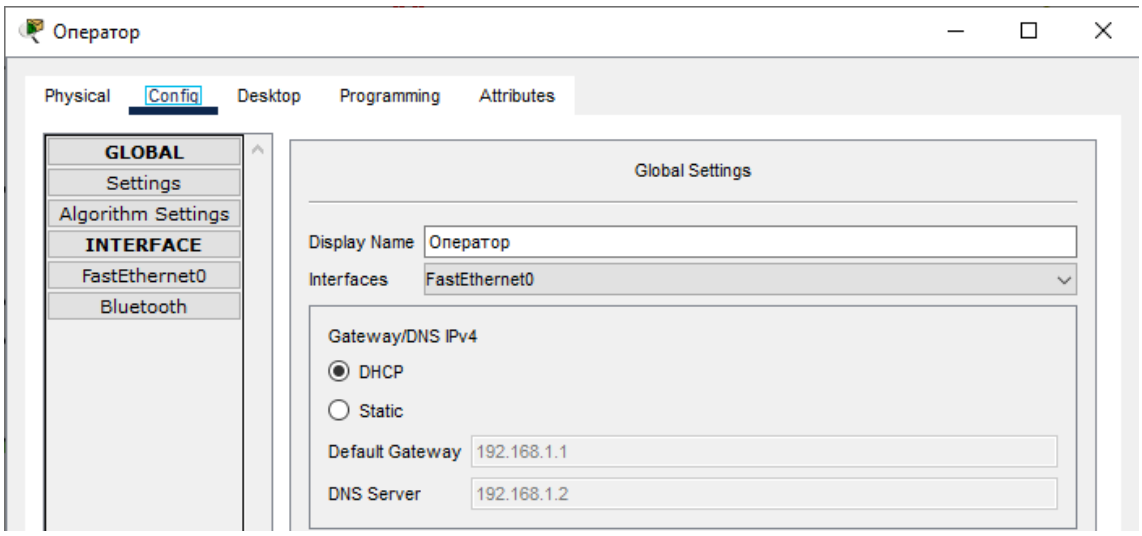


Рисунок 2.6 – Налаштування РМ оператора

Операторська станція може отримувати IP-адресу як автоматично (через DHCP), так і вручну, залежно від обраної конфігурації. Таке налаштування дозволяє оператору контролювати роботу IoT-пристроїв та адмініструвати систему з мінімальними затримками.

Лістинг 2.9 відображає налаштування сервера IoT, який є центральним елементом управління сенсорними пристроями та розумною автоматизацією у котеджному комплексі.

Лістинг 2.9 –Налаштування серверу IoT

```
Cisco Packet Tracer SERVER Command Line 1.0
C:\>ipconfig /all

GigabitEthernet0 Connection: (default port)
   Connection-specific DNS Suffix .:
   Physical Address . . . . . .: 0001.4203.8878
   Link-local IPv6 Address  . . .: FE80::201:42FF:FED3:8878
   IPv6 Address . . . . . .:
   IPv4 Address . . . . . .: 192.168.0.2
   Subnet Mask  . . . . . .: 255.255.255.0
   Default Gateway  . . . . . .: 192.168.0.1
   DHCP Servers . . . . . .: .
   DHCPv6 IAID  . . . . . .:
   DHCPv6 Client DUID . . . . . .: 00-01-00-01-AC-54-76-96-00-
01-42-D3-88-78
   DNS Servers  . . . . . .: 192.168.0.2

GigabitEthernet1 Connection:
   Connection-specific DNS Suffix .:
```

```
        Physical Address . . . . . .: 0010.1182.C410
        Link-local IPv6 Address  . . . .: FE80::210:11FF:FEB2:C410
(Примітка: зазвичай FE80)
        IPv6 Address . . . . . .:
        IPv4 Address . . . . . .: 192.168.1.2
        Subnet Mask  . . . . . .: 255.255.255.0
        Default Gateway  . . . . . .: .
        DHCP Servers . . . . . .: .
        DHCPv6 IAID  . . . . . .:
        DHCPv6 Client DUID . . . . . .: 00-01-00-01-AC-54-76-96-00-
01-42-03-88-7B
        DNS Servers  . . . . . .: 192.168.0.2

    C:\>
```

Сервер має дві мережеві карти (GigabitEthernet0 і GigabitEthernet1), що дозволяє йому працювати у двох VLAN-сегментах одночасно. Перша мережева карта отримала IP-адресу 192.168.0.2 та використовується у VLAN 100 для взаємодії із користувацькими пристроями. Друга мережева карта налаштована з IP-адресою 192.168.1.2 і працює у VLAN 101, що забезпечує підключення IoT-пристроїв та передачу даних між сенсорами.

Для кожного інтерфейсу налаштовано маску підмережі 255.255.255.0, що відповідає мережевому простору. Основний шлюз за замовчуванням для VLAN 100 вказано як 192.168.0.1, а для VLAN 101 – 192.168.1.1. Також сервер виконує функції DNS, що дозволяє пристроям IoT та користувачам коректно знаходити ресурси всередині мережі.

Завдяки такій конфігурації сервер IoT може керувати сенсорними пристроями, виконувати автоматизовані процеси та забезпечувати безперебійну взаємодію між усіма елементами розумної мережі котеджного комплексу.

2.3.6 Тестування мережі

Після завершення налаштування мережевої інфраструктури котеджного комплексу необхідно виконати комплексне тестування для перевірки функціональності всіх сегментів мережі, коректної роботи VLAN-сегментації та доступу клієнтів до мережевих сервісів. Основною метою тестування є переконатися у правильному функціонуванні DHCP-серверів, доступності

маршрутизатора для кінцевих пристроїв та коректному розподілі IP-адрес відповідно до налаштованих VLAN.

Одним із ключових аспектів перевірки є автоматична видача IP-адрес пристроям, підключеним до VLAN 100 та VLAN 101. Для цього необхідно перевірити DHCP-роздачу, яка забезпечує динамічне призначення IP-адрес для користувачів та IoT-пристроїв, а також гарантує, що пристрої отримують правильні конфігурації відповідно до визначених пулів. Наявність коректних записів у DHCP-сервері маршрутизатора свідчить про успішне налаштування сегментації мережі та доступу пристроїв.

На лістингу 2.10 наведено перевірку виданих IP-адрес через DHCP-сервер маршрутизатора, що дозволяє підтвердити наявність підключених клієнтів у двох VLAN. Відображений вивід команди show ip dhcp binding демонструє список клієнтів, яким автоматично видано IP-адреси разом із MAC-адресами їхніх пристроїв.

Лістинг 2.10 – Присутній доступ до маршрутизатора з обох VLAN

```
Router#sh ip dhcp bind

IP address      Client-ID/          Lease expiration        Type
                Hardware address
192.168.0.22    0060.3E09.EA73     --                   Automatic
192.168.1.24    0020.B0D0.AD8D     --                   Automatic
192.168.1.20    0007.ECEA.DAA4     --                   Automatic
192.168.1.23    0020.7F30.7708     --                   Automatic
192.168.1.24    0002.2F50.3400     --                   Automatic
192.168.1.25    000C.CFE6.C5DD     --                   Automatic
192.168.1.26    000C.85A0.634C     --                   Automatic
192.168.1.28    0001.6423.3349     --                   Automatic
192.168.1.31    0009.7C5D.4C49     --                   Automatic
192.168.1.29    0090.215D.043C     --                   Automatic
192.168.1.30    0070.2EDC.6C3C     --                   Automatic
192.168.1.27    000C.3E4C.E2DB     --                   Automatic
192.168.1.33    0060.5C17.B461     --                   Automatic
192.168.1.34    0090.2BAD.2405     --                   Automatic
192.168.1.36    0005.5E7E.1651     --                   Automatic
192.168.1.38    0000.5A41.8431     --                   Automatic
192.168.1.35    0060.3E25.8CB4     --                   Automatic
192.168.1.37    0090.7C16.2844     --                   Automatic
192.168.1.40    000A.41A9.1932     --                   Automatic
192.168.1.39    0090.2180.A5B2     --                   Automatic
192.168.1.42    0020.B2D0.B332     --                   Automatic
```

```
192.168.1.41    0000.A3EA.0C85     --                    Automatic
192.168.1.43    0001.9E64.C3D3     --                    Automatic
192.168.1.44    0000.B011.A981     --                    Automatic
192.168.1.45    0004.9AE8.938D     --                    Automatic
192.168.1.46    0030.A3C1.3417     --                    Automatic
192.168.1.49    000D.BD0D.42CC     --                    Automatic
192.168.1.48    0000.55E3.F951     --                    Automatic
192.168.1.50    0000.8E85.7E23     --                    Automatic
192.168.1.51    0001.42A6.B520     --                    Automatic
192.168.1.52    000C.8525.CAEA     --                    Automatic


Router#
```

Видно, що пристрої з VLAN 100 (користувачі) отримали адреси в діапазоні 192.168.0.x, а клієнти з VLAN 101 (IoT-пристрої) – у діапазоні 192.168.1.x. Усі IP-адреси розподіляються автоматично, що підтверджує стабільну роботу DHCP-сервера. Також видно, що MAC-адреси пристроїв записані в таблицю, що свідчить про успішне встановлення з’єднання між кінцевими пристроями та маршрутизатором.

Отримані результати свідчать про коректне налаштування VLAN-сегментації, DHCP-серверів та доступності маршрутизатора для всіх підключених клієнтів. Це гарантує, що кінцеві пристрої можуть успішно отримувати IP-адреси, взаємодіяти з сервером та передавати дані в межах заданої топології. Завдяки реалізованим налаштуванням мережа повністю готова до експлуатації та підтримує необхідний рівень продуктивності та безпеки.

У результаті виконаних етапів було створено повноцінну функціональну модель мережі, яка відповідає технічним та експлуатаційним вимогам проєкту. Реалізовано ефективну топологію з гіперканалами та VLAN-сегментацією, налаштовано централізоване керування точками доступу через WLC, а також впроваджено механізми динамічної адресації та ізоляції трафіку між користувачами й сенсорними пристроями. Проведене тестування підтвердило стабільну роботу всіх компонентів, коректний розподіл IP-адрес та доступність мережевих сервісів. Отримані результати засвідчують готовність проєктованої мережі до експлуатації у сучасному котеджному середовищі з підтримкою IoT та високим рівнем автоматизації.

3 НАЛАШТУВАННЯ РОБОТИ СЕНСОРНОЇ МЕРЕЖІ ТА МОНІТОРИНГ

3.1 Налаштування сенсорів розумної кімнати

Для забезпечення централізованого керування пристроями розумного дому в котеджному комплексі необхідно налаштувати реєстрацію IoT-пристроїв на сервері. Це дозволяє синхронізувати дані сенсорів, виконувати віддалене керування пристроями та забезпечувати стабільний обмін інформацією між компонентами системи. Усі IoT-пристрої підключаються до єдиного сервера, що гарантує централізовану обробку запитів та моніторинг стану мережі.

Підключення виконується через єдиний сервер IoT, розташований за адресою 192.168.0.2, що дозволяє здійснювати авторизацію та керування через встановлені параметри доступу. Налаштування підключення до серверу IoT зображене на рисунку 3.1.

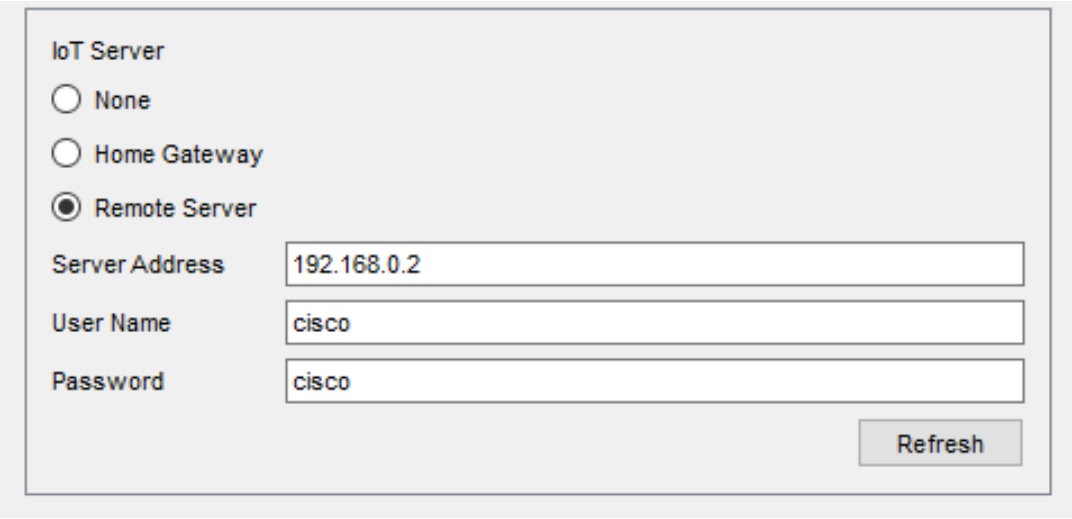


Рисунок 3.1 – Налаштування підключення до серверу IoT

На рисунку представлено інтерфейс конфігурації IoT-пристрою, який має три варіанти підключення:

– None – без підключення до центрального сервера;

– Home Gateway – локальний шлюз для управління в межах окремої підмережі;

– Remote Server – централізоване підключення до віддаленого сервера.

У даній конфігурації вибрано режим «Remote Server», що дозволяє IoT-пристроям підключатися до головного сервера за адресою 192.168.0.2. У відповідних полях введені облікові дані для авторизації: ім'я користувача «cisco» та пароль «cisco».

Після введення параметрів необхідно натиснути «Refresh», що дозволяє оновити стан підключення та перевірити успішність з'єднання з сервером. Завдяки цій конфігурації IoT-пристрої зможуть отримувати команди, передавати дані про свій стан та взаємодіяти з іншими елементами розумного будинку, забезпечуючи стабільну та безпечну роботу мережі.

Для централізованого управління та моніторингу всіх підключених IoT-пристроїв у котеджному комплексі використовуэться інтерфейс IoT-монітору на робочій машині оператора, який зображено на рисунку 3.2.

У вікні відображається повний список активних пристроїв, що були успішно зареєстровані на IoT-сервері. Серед них є термостати, кондиціонери, обігрівачі, двері, вікна, сирени та детектори руху. Кожен пристрій має унікальний ідентифікатор та статус підключення, що дозволяє оператору контролювати їхню роботу в реальному часі.

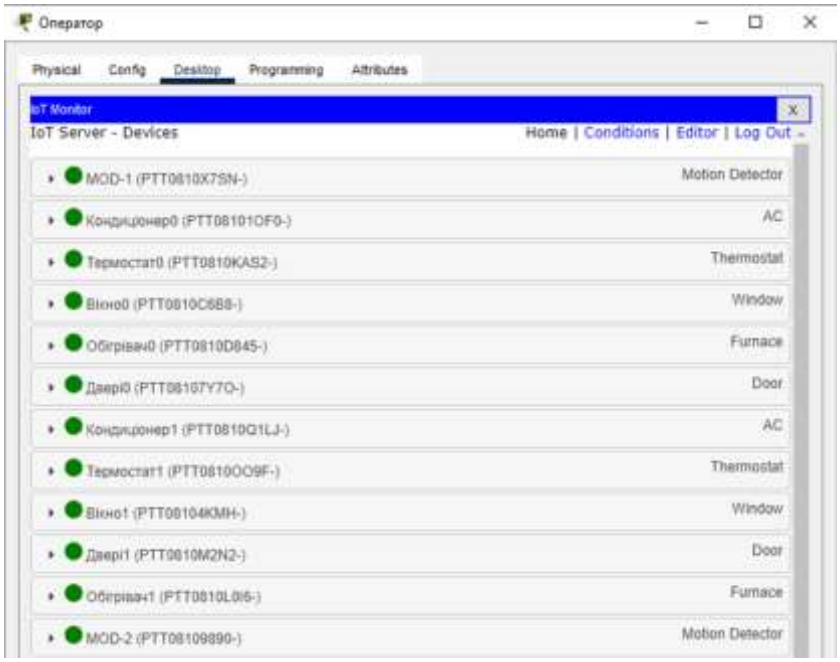


Рисунок 3.2 – Список IoT-приладів

Усі пристрої підключені до єдиного серверу IoT, що забезпечує автоматизоване управління системами розумного будинку. Завдяки цій конфігурації оператор може віддалено керувати обладнанням, переглядати стан пристроїв та змінювати параметри їхньої роботи, що підвищує ефективність функціонування всієї мережі.

Для прикладу Котеджу-0 виконаємо налаштування автоматичного вмикання та вимикання кондиціонера та обігрівача залежно від температури та стану вікон і дверей. Конфігурація відображена в таблиці 3.1.

Перше із 4-ох правил визначає увімкнення обігрівача, якщо температура в кімнаті опустилася нижче 15°C і вікно зачинене. Друге правило деактивує обігрівач у разі підвищення температури до 15°C або відкриття вікна, запобігаючи нераціональному використанню енергії.

Третє правило задає увімкнення кондиціонера, якщо температура в приміщенні перевищує або дорівнює 25°C, при цьому вікно та двері мають бути зачинені. Якщо ж температура падає нижче 25°C або користувач відкриває вікно чи двері, кондиціонер автоматично вимикається відповідно до четвертого правила.

Таблиця 3.1 – Налаштування розумної кімнати 0

Умова	Активна	Назва правила	Логіка умови	Дія
Yes	Так	Обігрівач Вкл	Усі умови виконуються (Match all):• Температура < 15.0 °C• Вікно0 вимкнене (false)	Set Heater0 On to true
Yes	Так	Обігрівач Викл	Будь-яка умова (Match any):• Температура ≥ 15.0 °C• Вікно0 увімкнене (true)	Set Heater0 On to false
Yes	Так	Кондиціонер Вкл	Усі умови виконуються (Match all):• Температура ≥ 25.0 °C• Вікно0 вимкнене (false)• Двері закриті (false)	Set AirConditioner0 On to true
Yes	Так	Кондиціонер Викл	Будь-яка умова (Match any):• Температура < 25.0 °C• Вікно0 увімкнене (true)• Двері відкриті (true)	Set AirConditioner0 On to false

Рисунок 3.3 відображає приклад автоматичного ввімкнення обігрівача в розумній кімнаті котеджу відповідно до налаштованих умов керування

мікрокліматом. Відповідно до заданих параметрів, термостат зафіксував температуру нижче 15°C, а вікно залишилося закритим, що відповідає умовам для запуску обігрівача.

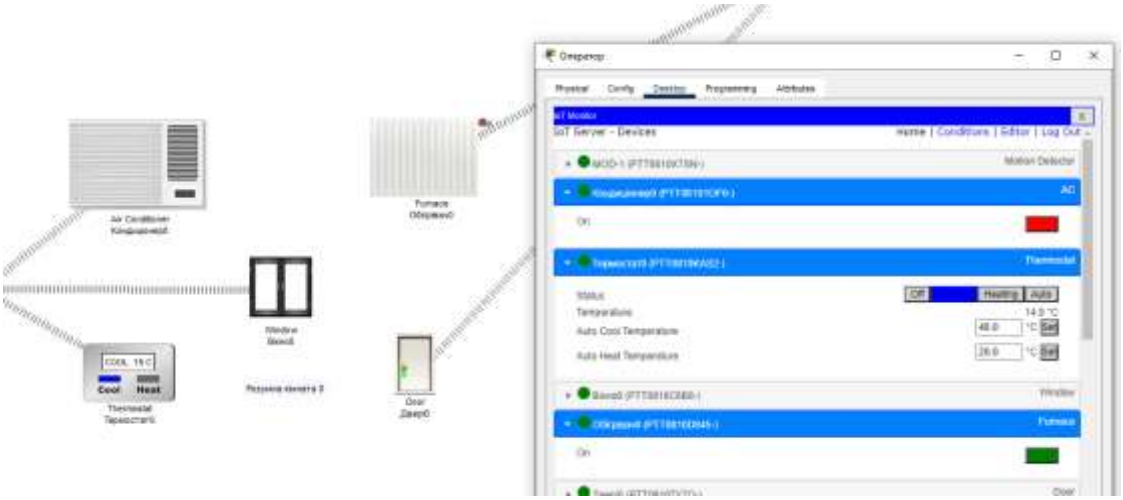


Рисунок 3.3 – Ввімкнення обігрівача

У вікні IoT-монітору можна бачити поточний статус пристроїв. Обігрівач отримав команду на ввімкнення, що підтверджується активним індикатором у моніторинговій системі. Водночас кондиціонер залишається вимкненим, оскільки температура не перевищує 25°C.

Система автоматичного контролю дозволяє оптимізувати мікроклімат у приміщенні, зменшуючи необґрунтоване використання енергоресурсів. Це забезпечує комфортні умови для мешканців та покращує загальну енергоефективність котеджного комплексу.

3.2 Налаштування запису подій

Для забезпечення безпеки у котеджному комплексі необхідно налаштувати автоматичне ввімкнення відеоспостереження при виявленні руху. Це дозволяє оптимізувати використання ресурсів системи, зменшити кількість записаних даних

та забезпечити оперативне реагування на події. Для реалізації такої функції використовується механізм правил (Conditions) на IoT-сервері, що дозволяє створювати тригерні події на основі роботи сенсорів.

При налаштуванні використовується детектор руху MOD-1, розташований у Містечку-1. При виявленні руху пристроєм передається сигнал на сервер, що активує камеру відеоспостереження (Webcam-1). Це забезпечує автоматизацію процесу без участі оператора та гарантує реєстрацію потенційних інцидентів у режимі реального часу. Детальна конфігурація представлена на рисунку 3.4.

На рисунку зображено налаштування правила «Motion-town-1», яке визначає, коли камера повинна автоматично вмикатися. У полі «If» задається умова спрацювання тригера – детектор руху MOD-1 має бути «On», що означає виявлення руху.

Рисунок 3.4 – Умова для ввімкнення камери на Містечко-1

У полі «Then set» визначається автоматична дія у відповідь – при виконанні умови необхідно ввімкнути камеру Webcam-1 у стан «On». Це дозволяє вести запис лише у момент активності, що оптимізує використання пам’яті та зменшує навантаження на сервер відеоспостереження.

Рисунок 3.5 відображає налаштування умови для автоматичного вимкнення відеоспостереження у Містечку-1. Ця конфігурація є логічним продовженням попереднього правила та дозволяє оптимізувати роботу системи відеоспостереження, вимикаючи камеру після зникнення руху, що знижує навантаження на мережу та сервер збереження відеозаписів.

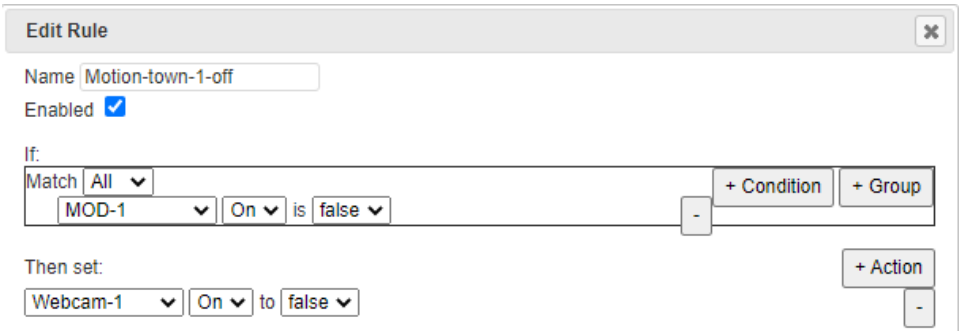


Рисунок 3.5 – Умова для вимкнення камери на Містечко-1

Правило «Motion-town-1-off» визначає умову завершення запису. У полі «If» вказано, що детектор руху MOD-1 змінює стан на false (тобто, рух більше не виявлено). У полі «Then set» задається дія – перевести Webcam-1 у стан false, що автоматично зупиняє запис відео.

На рисунку 3.6 наведено список налаштованих правил автоматичного керування відеоспостереженням у котеджному комплексі. Всього створено чотири правила, які відповідають за ввімкнення та вимкнення камер відеоспостереження у двох містечках.

Actions		Enabled	Name	Condition	Actions
Edit	Remove	Yes	Motion-town-1	MOD-1 On is true	Set Webcam-1 On to true
Edit	Remove	Yes	Motion-town-2	MOD-2 On is true	Set Webcam-2 On to true
Edit	Remove	Yes	Motion-town-1-off	MOD-1 On is false	Set Webcam-1 On to false
Edit	Remove	Yes	Motion-town-2-off	MOD-2 On is false	Set Webcam-2 On to false

Рисунок 3.6 – Список налаштованих правил керування відеоспостереженням

Перші два правила «Motion-town-1» та «Motion-town-2» активують відповідні камери Webcam-1 та Webcam-2, якщо сенсори MOD-1 або MOD-2 фіксують рух. Два наступні правила «Motion-town-1-off» та «Motion-town-2-off» забезпечують автоматичне вимкнення камер після припинення руху.

Завдяки реалізованій конфігурації система відеоспостереження працює оптимально, активуючи запис лише в моменти необхідності. Це мінімізує використання ресурсів мережі та пам’яті, а також дозволяє швидко аналізувати відеоархів у разі необхідності.

На рисунку 3.7 зображено процес тестування роботи детектора руху та його інтеграції з системою відеоспостереження. Після виявлення руху сенсором Motion Detector MOD-1 відбувається автоматичне ввімкнення камери Webcam-1, що підтверджує коректне функціонування налаштованих тригерних правил.

На зображенні видно, що детектор руху активований, що сигналізує про його спрацьовування. Відповідно, камера переходить у режим запису, що візуально відображається на її індикаторі. Це підтверджує, що система безпеки працює відповідно до запрограмованих умов, а відеофіксація активується лише у разі необхідності.

Даний механізм дозволяє оптимізувати роботу відеоспостереження, скорочуючи обсяг непотрібних записів та зменшуючи навантаження на систему збереження відео. Тестування показало, що система швидко реагує на зміну стану сенсора, гарантуючи оперативне ввімкнення відеоспостереження у разі виявлення руху.

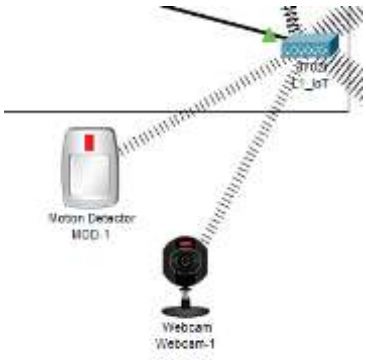


Рисунок 3.7 – Спрацьовування детектору руху та ввімкнення камери

3.3 Налаштування аварійного сповіщення

Аварійна система сповіщення є критично важливим елементом інфраструктури котеджного комплексу, що забезпечує оперативне інформування мешканців про надзвичайні ситуації. Основними тригерами для активації

аварійного сигналу є натискання кнопки аварійного сповіщення або перевищення допустимої швидкості вітру, що може свідчити про небезпечні погодні умови.

Для реалізації цього функціоналу до мережі підключено контролер MCU, який виконує збір даних з датчиків та керує активацією сирен. До цифрового входу D0 контролера підключена кнопка аварійного сповіщення, а до аналогового входу A0 — датчик вітру, що дозволяє в автоматичному режимі контролювати зміну погодних умов. Детальна схема підключення представлена на рисунку 3.8.

На рисунку зображено структуру взаємодії пристроїв у системі аварійного оповіщення. Контролер MCU-0 виконує роль центрального вузла керування. Він отримує сигнал від кнопки аварійного сповіщення та дані від датчика вітру. При досягненні порогу у 512 умовних одиниць, що свідчить про сильний вітер, контролер передає команду на активацію сирени.

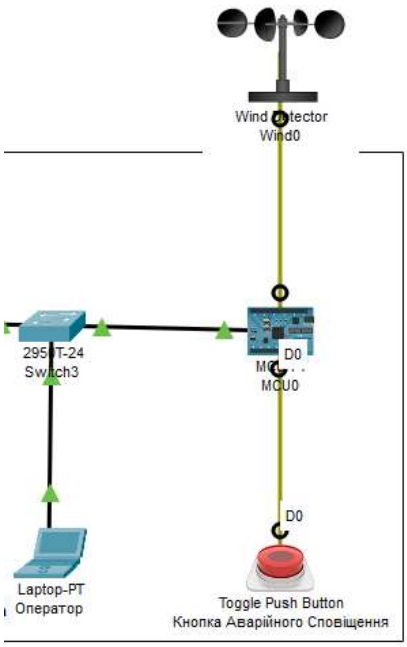


Рисунок 3.8 – Підключення входів до MCU

Зв'язок між контролером та іншими елементами системи здійснюється через підключення до VLAN 101, що забезпечує стабільний зв’язок та швидке реагування на критичні події. Оператор може спостерігати стан пристроїв у реальному часі, а у разі необхідності активувати або деактивувати систему вручну.

Програмний код контролера MCU для реалізації автоматичного аварійного сповіщення наведено в Додатку А. Код написаний мовою JavaScript і реалізує логіку моніторингу стану кнопки аварійного сповіщення та показників датчика вітру.

Функції setup виконує налаштування вхідних пінів для підключених пристроїв. Кнопка аварійного сповіщення зчитується через цифровий вхід D0, а показники датчика вітру отримуються з аналогового входу A0. Значення цих параметрів передаються у вигляді логічного сигналу (true/false) для кнопки та числового значення для сили вітру.

У циклі loop постійно відбувається перевірка змін стану кнопки та значень датчика. Якщо натискання на кнопку фіксується, змінюється стан перемінної state, а якщо сила вітру змінюється – оновлюється wind. Передача даних на сервер IoT здійснюється тільки при зміні стану, що дозволяє оптимізувати обмін інформацією та мінімізувати навантаження на мережу.

3.4 Тестування роботи сенсорів

Для перевірки коректності функціонування сенсорної мережі необхідно провести тестування роботи всіх підключених датчиків, що відповідають за контроль навколишніх умов та безпеку мешканців котеджного комплексу. Основною метою тестування є оцінка швидкості реакції системи на критичні події, перевірка коректності передачі даних між сенсорами та IoT-сервером, а також аналіз автоматичного виконання заданих сценаріїв.

Особливу увагу під час тестування приділяється системі аварійного сповіщення, яка є одним із ключових компонентів безпеки комплексу. Для перевірки її роботи необхідно змоделювати різні сценарії, зокрема натискання кнопки тривоги та штучне підвищення швидкості вітру вище порогового значення.

На рисунку 3.9 зображено процес тестування роботи аварійного сповіщення у котеджному комплексі. Активація сирени відбулася у відповідності до заданих умов – або після натискання кнопки аварійного сповіщення, або у разі перевищення порогового значення швидкості вітру.

У IoT-моніторі відображено поточний стан пристроїв, де MCU-0 зафіксував подію, а сирена Sire-2 перейшла у режим увімкнення. Це підтверджує коректне функціонування контролера та успішну передачу сигналу через мережу IoT.

Фізична схема демонструє зв'язок між датчиком вітру, кнопкою аварійного сповіщення, контролером MCU та мережею IoT. Система працює у реальному часі та забезпечує миттєве реагування на критичні події.

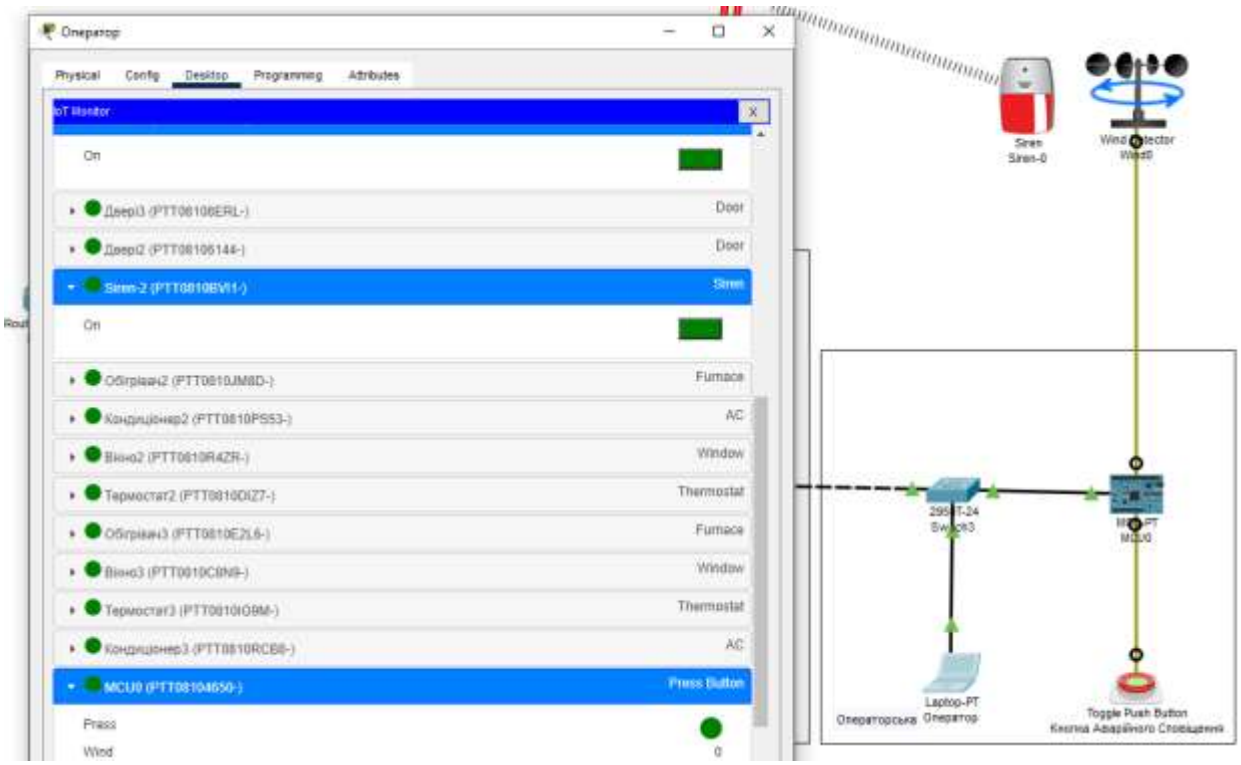


Рисунок 3.9 – Аварійне сповіщення ввімкнене на усіх майданчиках

Результати тестування показали, що аварійне сповіщення активується на всіх майданчиках одночасно, що є важливою умовою для забезпечення безпеки мешканців котеджного комплексу.

3.5 Налаштування доступу до мережі інтернет для серверної частини та користувацької мережі

Для забезпечення стабільного та безпечного доступу до глобальної мережі Інтернет у котеджному комплексі необхідно виконати налаштування NAT (Network Address Translation) для обох VLAN. Це дозволить перетворювати локальні IP-адреси у зовнішні при виході у глобальну мережу, а також контролювати вхідний трафік відповідно до встановлених політик безпеки.

Налаштування NAT виконується на маршрутизаторі, де необхідно задати інтерфейси для зовнішнього та внутрішнього трафіку, створити список контролю доступу (ACL), а також налаштувати правила трансляції для обох VLAN. Це дозволить розмежувати доступ між користувачами та IoT-пристроями, забезпечуючи оптимальний баланс безпеки та продуктивності мережі. Детальна конфігурація NAT наведена в лістингу 3.1.

Лістинг 3.1 – Конфігурація NAT

```
interface GigabitEthernet0/0
 ip address 8.8.8.2 255.255.255.252
 ip nat outside
!
…
!
interface GigabitEthernet1/0.100
 encapsulation dot1Q 100
 ip address 192.168.0.1 255.255.255.0
 ip nat inside
!
interface GigabitEthernet1/0.101
 encapsulation dot1Q 101
 ip address 192.168.1.1 255.255.255.0
 ip nat inside
!
ip nat inside source list internet interface GigabitEthernet0/0
overload
!
ip access-list extended internet
```

Через лістинг відображено процес конфігурування NAT, що включає призначення інтерфейсу GigabitEthernet0/0 як зовнішнього (NAT outside), а інтерфейсів GigabitEthernet1/0.100 та GigabitEthernet1/0.101 як внутрішніх (NAT inside). Також створено список контролю доступу (ACL) для визначення дозволених підмереж, що можуть використовувати NAT, і реалізовано перетворення вихідного трафіку через відповідний інтерфейс.

У лістингу 3.2 представлено тестування доступу до мережі Інтернет зі смартфона користувача. Для перевірки коректності роботи NAT і маршрутизації виконано трасування маршруту до імітатора мережі Інтернет. Відповідно до налаштованих політик, трафік проходить через внутрішній шлюз, після чого транслюється у зовнішню мережу.

Лістинг 3.2 – Трасування маршруту до імітатора мережі інтернет

```
C:\>ipconfig /all

Wireless0 Connection: (default port)
   Connection-specific DNS Suffix
   Physical Address           00-60-3E-09-EA-73
   Link-local IPv6 Address    FE80::260:3EFF:FE09:EA73
   IPv6 Address               (не вказано)
   IPv4 Address               192.168.0.22
   Subnet Mask                255.255.255.0
   Default Gateway            192.168.0.1
   DHCP Servers               192.168.0.1
   DHCPv6 IAID                1067502149
   DHCPv6 Client DUID          00-01-00-01-E0-30-E9-CE-00-60-3E-
09-EA-73
   DNS Servers                192.168.0.2

3G/4G Cell Connection:
   Connection-specific DNS Suffix
   Physical Address           00-00-0C-DA-A6-99
   Link-local IPv6 Address    FE80::200:CFF:FEDA:A699

C:\>tracert 8.8.8.8

Tracing route to 8.8.8.8 over a maximum of 30 hops:

  1    44 ms     7 ms    22 ms  192.168.0.1
  2    16 ms    36 ms     2 ms  8.8.8.8

Trace complete.
C:\>
```

Результати тестування підтверджують стабільне підключення користувача до глобальної мережі та правильність виконаної конфігурації NAT на маршрутизаторі. Це свідчить про успішне налаштування доступу до Інтернету для клієнтських пристроїв у котеджному комплексі.

Завдяки такій конфігурації користувачі та IoT-пристрої отримують захищений доступ до Інтернету без необхідності безпосереднього використання глобальних IP-адрес, що підвищує рівень безпеки та ефективність використання мережевих ресурсів у котеджному комплексі.

3.6 Вибір алгоритмів машинного навчання для аналізу та прогнозування стану мережі

Для забезпечення надійного аналізу та прогнозування стану мережевої інфраструктури котеджного комплексу необхідно застосовувати методи машинного навчання, здатні швидко обробляти потоки даних та виявляти потенційні несправності на основі історичних записів. Вибір алгоритмів базується на критеріях швидкодії, точності прогнозування та можливості ефективного навчання на обмежених вибірках. Особливу увагу приділяється методам, які дозволяють працювати з деревоподібними структурами рішень, що добре підходять для класифікації стану мережевих пристроїв та прогнозування відмов.

Швидке дерево рішень (ШДР) – це ефективний метод, що базується на ансамблевій моделі градієнтного бустингу. Алгоритм поєднує в собі гнучкість дерев рішень та швидкість обчислень, що дозволяє швидко обробляти великі обсяги даних про стан мережі. Завдяки механізмам регуляризації та адаптивного навчання, ШДР здатний виявляти аномалії у функціонуванні мережевих пристроїв та прогнозувати ймовірність їхньої відмови.

Випадковий ліс (ВЛ) – це ансамблевий метод, що використовує множину незалежних дерев рішень для підвищення точності прогнозування. Його основною

перевагою є стійкість до шумових даних та здатність працювати з великою кількістю вхідних параметрів, що є критично важливим для аналізу складної мережевої інфраструктури. ВЛ добре підходить для класифікації стану мережевих компонентів, дозволяючи визначати потенційно проблемні сегменти без значного впливу поодиноких некоректних вимірювань.

Дерева рішень (ДР) – ще один швидкодіючий метод, що поєднує переваги градієнтного бустингу та оптимізованих дерев рішень. Основна особливість ДР – ефективна робота з великими потоками даних та високошвидкісне навчання, що дозволяє використовувати його для реального моніторингу стану мережі. Завдяки ефективному використанню пам’яті та можливості обробки великої кількості параметрів, ДР забезпечує високу продуктивність у виявленні аномалій та прогнозуванні відмов мережевих пристроїв.

Для визначення оптимального методу аналізу та прогнозування стану мережі важливо порівняти їх за ключовими характеристиками, які впливають на продуктивність і точність роботи. Основними критеріями вибору є швидкість навчання, обчислювальна ефективність, стійкість до шумових даних, здатність до роботи з великими наборами ознак та точність прогнозування. Наведені алгоритми мають різні підходи до обробки даних, що впливає на їхню ефективність у різних сценаріях застосування.

У таблиці 3.2 наведено порівняння розглянутих алгоритмів за основними характеристиками, що є критично важливими для моніторингу мережевої інфраструктури та виявлення можливих збоїв.

Таблиця 3.2 – Порівняння методів машинного навчання

Характеристика	ШДР	ВЛ	ДР
Швидкість навчання	Висока	Середня	Висока
Обчислювальна ефективність	Висока	Середня	Висока
Точність прогнозування	Висока	Висока	Висока
Стійкість до шумових даних	Висока	Висока	Середня
Здатність до роботи з великими наборами ознак	Висока	Середня	Висока

Вибір методу ШДР для розробки системи моніторингу мережі обумовлений його високою швидкістю навчання, ефективністю роботи з великими наборами ознак та стійкістю до шумових даних. Завдяки оптимізованому механізму бустингу, алгоритм дозволяє швидко навчатися навіть на великих обсягах мережевого трафіку, що критично важливо для реального часу. Висока обчислювальна ефективність дозволяє зменшити навантаження на серверні ресурси, забезпечуючи оперативний аналіз даних без втрати продуктивності. Порівняно з іншими методами, ШДР краще адаптується до складних закономірностей у поведінці мережі, що підвищує точність прогнозування можливих збоїв і дозволяє своєчасно реагувати на критичні події. Це робить алгоритм найбільш придатним для впровадження у систему інтелектуального моніторингу мережевої інфраструктури котеджного комплексу.

3.7 Попередня обробка даних та особливості роботи з датасетом

Процес побудови системи машинного навчання для аналізу та прогнозування стану мережі котеджного комплексу базується на ретельній обробці вхідних даних. Для забезпечення високої точності моделі та стабільності її роботи необхідно усунути можливі недоліки вхідного датасету, такі як пропущені значення, нерівномірний розподіл класів або зайві ознаки. Попередня обробка включає кілька етапів, кожен з яких спрямований на покращення якості даних та підготовку їх до подальшого навчання моделі. Основні етапи цього процесу наведено на рисунку 3.10.

Структура побудови моделі машинного навчання включає чотири ключові етапи розробки моделі машинного навчання. Першим кроком є підготовка даних, що включає очищення вхідних записів, інтеграцію даних із різних джерел та їх балансування. Очищення даних охоплює обробку викидів, видалення дублювань і нормалізацію значень для забезпечення коректної роботи алгоритму. Далі

виконується інженерія ознак, що передбачає витягнення нових інформативних характеристик та відбір найбільш значущих ознак для навчання моделі.

Наступний крок – вибір та навчання моделі, який включає визначення найбільш придатного алгоритму, оптимізацію його гіперпараметрів та процес навчання на тренувальному наборі даних. У цьому етапі також здійснюється крос-валідація для покращення узагальнюючої здатності моделі та мінімізації ризику перенавчання. Завершальним кроком є оцінювання продуктивності моделі та її впровадження у робоче середовище. Проводиться тестування на незалежному тестовому наборі даних, аналізуються основні метрики продуктивності (точність, F1-score, повнота), після чого модель інтегрується у реальну систему для прогнозування стану мережі.

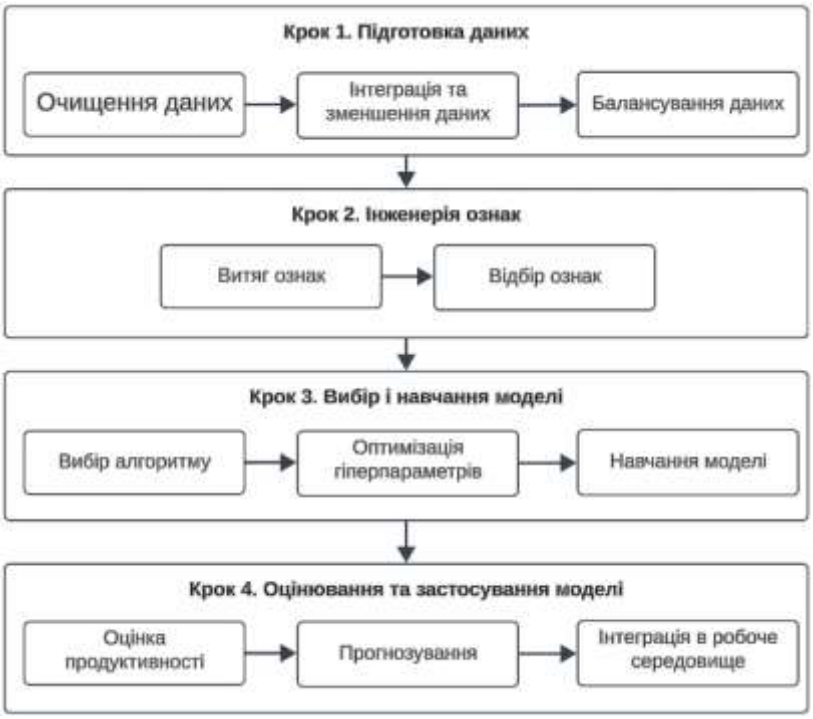


Рисунок 3.10 – Структура побудови моделі машинного навчання

Для задачі аналізу стану мережі та прогнозування можливих несправностей було обрано спеціалізований датасет, що містить ключові показники роботи мережевого обладнання [21]. Ці дані включають параметри, які безпосередньо впливають на функціонування інфраструктури, зокрема рівень завантаженості мережі, кількість активних підключень, затримки передачі пакетів та інші

характеристики. Завдяки структурованості та наявності чітко визначених міток, цей набір даних ідеально підходить для задач класифікації та побудови прогнозної моделі.

Після вибору датасету було проведено його попередню обробку, що включала кілька ключових етапів. Спочатку було виконано очищення даних, яке передбачало виявлення та усунення шумових значень, що могли спотворити результати аналізу. Далі було здійснено роботу з пропущеними значеннями: вони або заповнювалися на основі статистичних методів, або видалялися, якщо їх частка була значною. Крім того, всі параметри було приведено до єдиного формату, що забезпечило узгодженість структури даних і спростило їх подальшу обробку. Така ретельна підготовка дозволила усунути потенційні проблеми, які могли б негативно вплинути на якість навчання моделі, а також забезпечила коректність аналізу та узгодженість ознак. У результаті отримано структурований набір даних, що містить тільки релевантні параметри, необхідні для ефективної побудови прогнозної системи.

У таблиці 3.3 наведено перелік основних параметрів датасету, що використовувалися для побудови аналітичної моделі. Кожен атрибут відіграє ключову роль у процесі аналізу даних, сприяючи виявленню закономірностей та прогнозуванню можливих збоїв у роботі системи. Завдяки чітко структурованим параметрам, модель може ефективно розрізняти стани мережевого обладнання та приймати відповідні рішення.

Таблиця 3.3 – Основні атрибути датасету

№	Атрибут	Опис
1	Unique ID	Унікальний ідентифікатор кожного запису, що дозволяє точно співвідносити дані.
2	Device ID	Ідентифікатор пристрою, що генерує дані, використовується для контролю джерел інформації.
3	Network Load (%)	Поточне завантаження мережі, що визначає рівень навантаження на інфраструктуру.
4	Packet Delay (ms)	Час затримки передавання пакетів у мілісекундах, впливає на якість зв’язку.
5	Signal Strength (dBm)	Потужність сигналу мережі, яка є критичним параметром для стабільності підключення.

Продовження таблиці 3.3

№	Атрибут	Опис
6	Connection Type	Тип підключення (дротове або бездротове), що визначає характеристики мережевої взаємодії.
7	Error Rate (%)	Відсотковий рівень помилок у переданих пакетах, що свідчить про надійність каналу зв’язку.
8	Data Throughput (Mbps)	Пропускна здатність каналу передачі даних, важливий параметр для аналізу продуктивності.
9	System Status	Мітка класу, що вказує на стан мережевого обладнання: нормальний (0) або аварійний (1).

Ці параметри забезпечують усебічне уявлення про стан мережевої інфраструктури, дозволяючи ефективно будувати прогнозні моделі. Завдяки їх комплексному аналізу можна визначати приховані закономірності, що впливають на стабільність роботи системи, та вчасно реагувати на можливі відхилення.

3.8 Тестування програмного забезпечення для забезпечення роботи мережі

У межах проєкту з проєктування мережі передачі даних котеджного комплексу із застосуванням стандартів сенсорних мереж було розроблено спеціалізоване програмне забезпечення для моніторингу та прогнозування можливих відмов у мережевій інфраструктурі. Основним призначенням цієї програми є підтримка безперебійної роботи критичних вузлів мережі шляхом аналізу телеметричних показників, що надходять із сенсорної інфраструктури комплексу. Програма забезпечує обробку вхідних даних про температуру, навантаження, швидкість обертання елементів, зношення обладнання тощо з метою прогнозного визначення технічного стану мережі – «нормального» або «аварійного». Таким чином, система виконує функцію інтелектуального моніторингу, здатного виявити потенційні відмови до їх фактичного настання, що значно підвищує рівень експлуатаційної безпеки комплексу.

Алгоритмічна основа рішення побудована на класифікаційній моделі машинного навчання, що дозволяє адаптивно аналізувати як поточні, так і історичні дані про функціонування мережі. Основний акцент у розробці було зроблено на забезпечення високої точності прогнозів при різних сценаріях експлуатації, що дозволяє системі формувати своєчасні рекомендації для технічного персоналу.

Код програмного рішення, реалізованого на мові C# з використанням бібліотеки ML.NET, наведено у Додатку Б. Він включає повний цикл роботи системи: завантаження моделі, обробку вхідних даних, виконання прогнозу стану, а також виведення результатів із відповідним рівнем упевненості. Особливу увагу приділено модульності архітектури та можливості масштабування системи, що є критично важливим у контексті поступового розширення сенсорної інфраструктури або підключення нових кластерів мережі.

Програмне забезпечення передбачає обробку даних у реальному часі, що дозволяє оперативно реагувати на зміни стану обладнання. Для підвищення достовірності результатів передбачено механізми нормалізації вхідних ознак, а також автоматичну обробку пропущених або аномальних значень, що можуть виникати під час збору інформації з фізичних сенсорів. Результати обробки відображаються у вигляді текстових повідомлень та можуть інтегруватися з іншими інформаційними панелями або системами диспетчерського контролю.

Після виконання всіх етапів підготовки даних, вибору моделі та її навчання було здійснено оцінку продуктивності алгоритму. Проведене навчання алгоритму дозволило перевірити його здатність ефективно розпізнавати як нормальний стан мережевої інфраструктури, так і відхилення, що можуть свідчити про потенційні збої в її роботі. На рисунку 3.11 наведено підсумкові результати навчання моделі, що містять ключові метрики оцінки її ефективності.

Оцінка продуктивності моделі здійснювалася за основними метриками, що характеризують її здатність правильно класифікувати позитивні та негативні зразки. Детальний аналіз точності (Precision), повноти (Recall) та F1-міри надав

можливість об'єктивно оцінити баланс між кількістю правильних позитивних прогнозів та здатністю моделі охопити всі реальні випадки відмов.

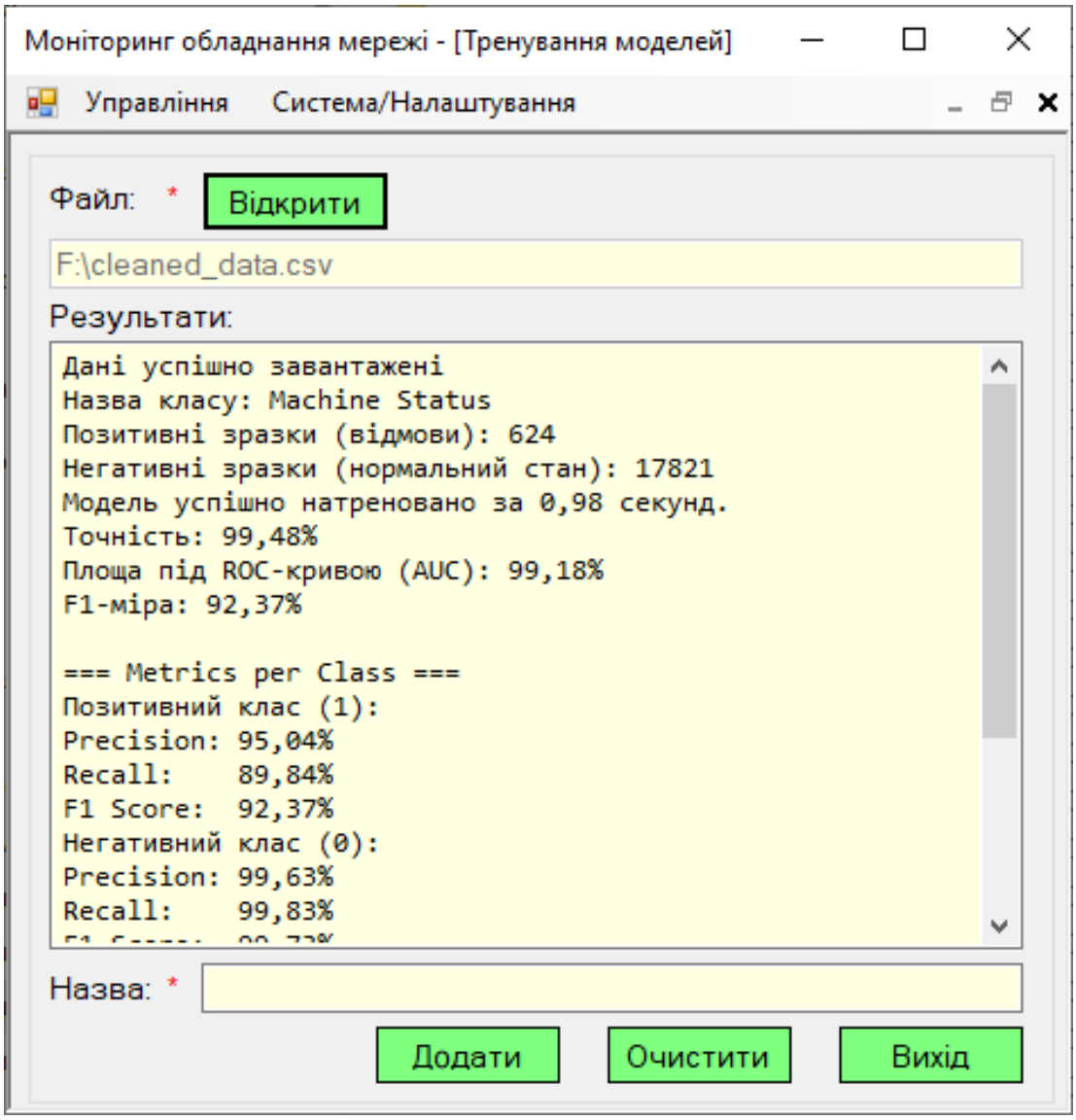


Рисунок 3.11 – Результат проведеного навчання моделі

Розуміння цих показників є фундаментальним для визначення надійності моделі у розрізненні штатного режиму роботи від передвісників збоїв, що безпосередньо впливає на ефективність превентивного обслуговування.

На рисунку 3.12 детально представлено кількісні значення цих метрик для кожного класу, що дозволить зробити вичерпні висновки про її практичну придатність.

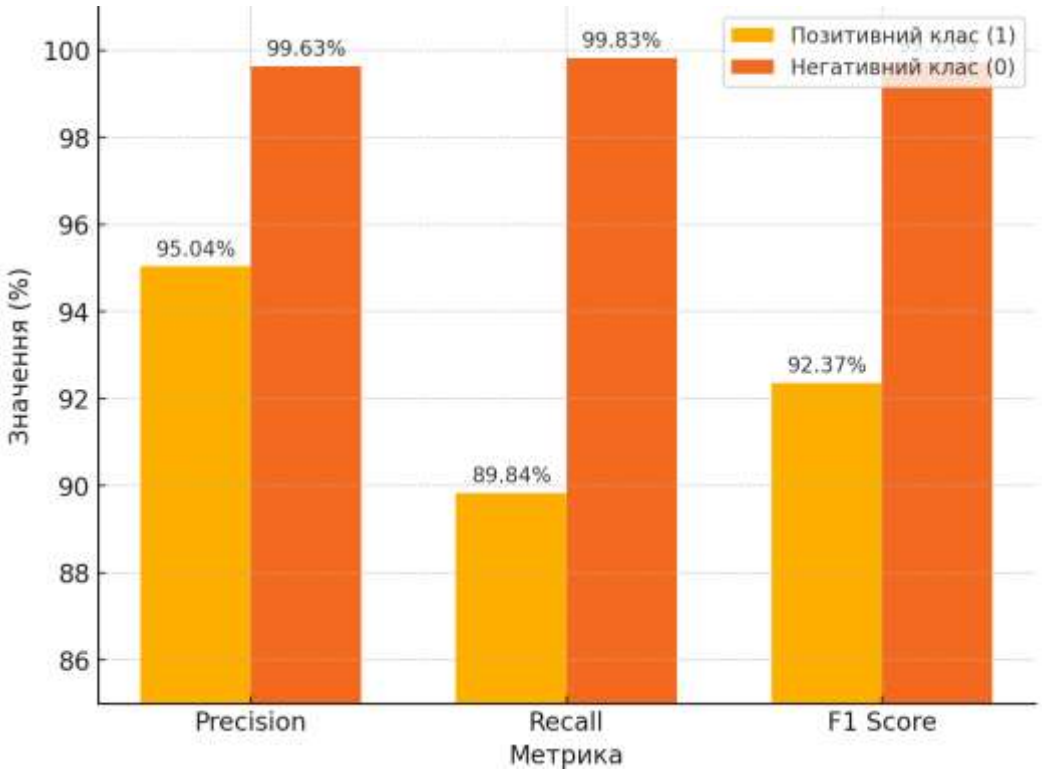


Рисунок 3.12 – Отримані метрики моделі для 2 класів

Отримані метрики свідчать про високу ефективність моделі у визначенні як нормального стану обладнання, так і його відмов. Для негативного класу (нормальний стан) значення Precision (99,63%) і Recall (99,83%) майже ідеальні, що вказує на мінімальну кількість помилкових спрацьовувань. Для позитивного класу (відмови) Precision досяг 95,04%, що означає, що більшість передбачених відмов дійсно були правильними, однак Recall у 89,84% вказує на можливі пропущені випадки несправностей. F1-міра 92,37% для відмов демонструє хороший баланс між точністю та повнотою, що є прийнятним для практичного використання. Загалом, модель добре справляється з класифікацією, але можлива подальша оптимізація для покращення Recall у виявленні відмов.

Отримані результати класифікації можна детальніше проаналізувати за допомогою матриці помилок, наведеної на рисунку 3.13. Вона дозволяє оцінити правильність передбачень моделі та кількість помилкових класифікацій для кожного з класів. У результаті тестування було отримано 115 правильно передбачених відмов обладнання (True Positives) та 3495 випадків коректного визначення нормального стану (True Negatives).

Водночас, модель допустила 6 помилкових позитивних спрацьовувань (False Positives), що означає некоректне виявлення відмови в обладнанні, яке насправді було справним. Крім того, було зафіксовано 13 випадків (False Negatives), коли модель не виявила несправності, хоча вони фактично відбулися. Такий розподіл помилок свідчить про високу загальну точність моделі, однак для зниження кількості пропущених відмов може бути доцільним додаткове налаштування чутливості алгоритму.

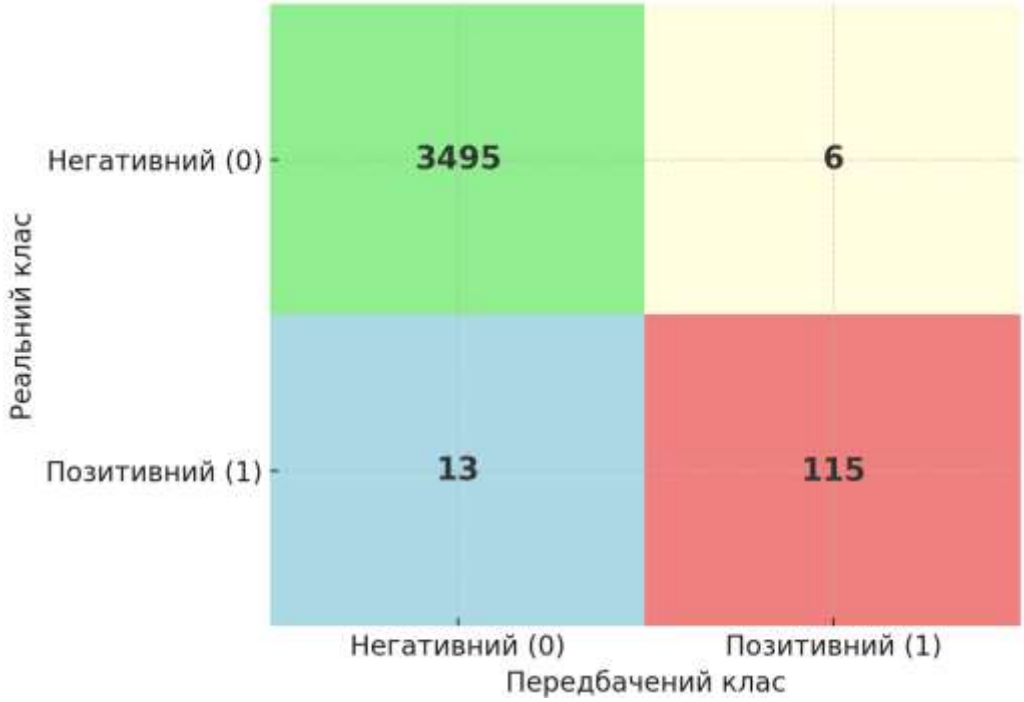


Рисунок 3.13 – Матриця помилок

Під час тестування моделі було виконано прогнозування на основі конкретного сценарію експлуатації обладнання, представленого на рисунку 3.14.

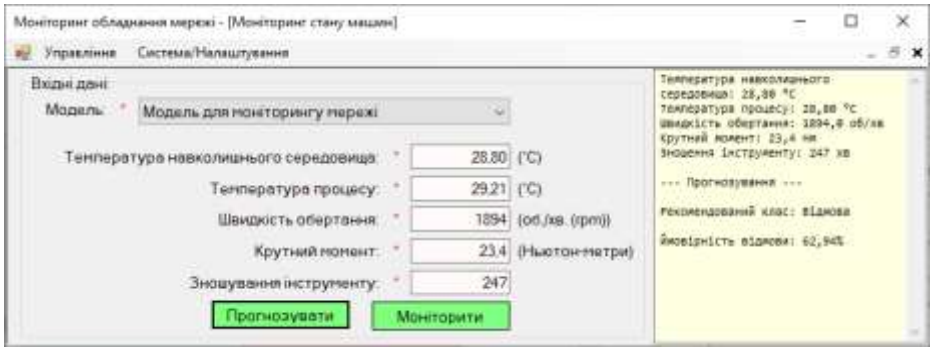


Рисунок 3.14 – Результати тестування моделі на 1-му сценарії

Вхідні параметри включали температуру навколишнього середовища 28,80 °C, температуру процесу 29,21 °C, швидкість обертання 1894,0 об/хв, крутний момент 23,4 Нм та рівень зношення інструменту 247 хв. Аналіз цих значень дозволив моделі визначити можливий стан обладнання. В результаті прогнозування модель віднесла даний випадок до класу «Відмова» із ймовірністю 62,94%. Це свідчить про те, що вхідні параметри демонструють потенційну небезпеку несправності, що дозволяє оператору вжити превентивних заходів для уникнення аварійної ситуації.

У наступному тестовому сценарії, наведеному на рисунку 3.15, модель отримала інші параметри експлуатації обладнання для оцінки його стану.

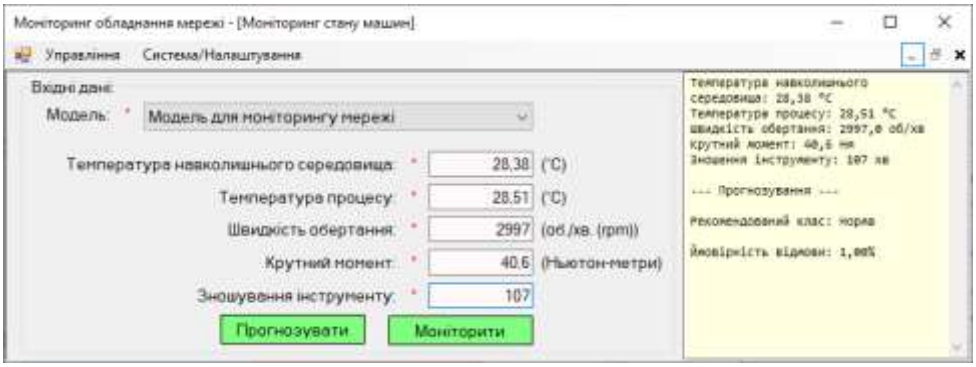


Рисунок 3.15 – Тестування моделі на 2-му сценарії

Вхідні значення включали температуру навколишнього середовища 28,38 °C, температуру процесу 28,51 °C, швидкість обертання 2997,0 об/хв, крутний момент 40,6 Нм та зношення інструменту 107 хв. Аналіз цих характеристик дозволив моделі класифікувати стан обладнання як «Норма». Ймовірність відмови, розрахована системою, склала лише 1,00%, що вказує на стабільний та безпечний режим роботи. Отримані результати підтверджують коректність роботи моделі у випадках, коли система функціонує в межах нормальних експлуатаційних параметрів.

У результаті виконаних етапів було повністю налаштовано сенсорну інфраструктуру котеджного комплексу, включно з підключенням розумних пристроїв до серверного середовища, налаштуванням умов для керування

мікрокліматом, організацією автоматизованого відеоспостереження та впровадженням системи аварійного сповіщення. Крім того, реалізовано конфігурацію NAT для безпечного доступу до Інтернету та інтегровано інтелектуальне програмне забезпечення для аналізу й прогнозування стану мережі на основі алгоритмів машинного навчання. Проведене тестування підтвердило правильність роботи конфігурацій та ефективність побудованої архітектури. Таким чином, отримано функціональну систему, що повністю відповідає вимогам проєкту й забезпечує стабільну та безпечну експлуатацію сенсорної мережі в умовах розгалуженого котеджного середовища.

ВИСНОВКИ

Сучасна мережа передачі даних у котеджному комплексі з сенсорними технологіями необхідна для надійного зв’язку між елементами інфраструктури (IoT-пристроями, системами безпеки, точками доступу). Вона забезпечує безперервну взаємодію між вузлами та централізоване управління, що є критично важливим для масштабованих житлових середовищ.

При проєктуванні враховано технічні потреби мешканців і специфіку розміщення об’єктів. Для бездротового доступу обрано Wi-Fi 6, що забезпечує високу швидкість передачі даних і підтримку численних клієнтських пристроїв. У сенсорному сегменті мережі використано енергоефективні NB-IoT і ZigBee, оптимальні для підключення численних малопотужних датчиків (сенсори температури, руху, відеонагляду).

Для гнучкості мережі та спрощення розгортання пристроїв впроваджено динамічну IP-адресацію, що уникає ручної конфігурації адрес. Внутрішній трафік захищено NAT, а доступ на критичних сегментах контролюється ACL. Підтримка протоколу 802.1Q дозволила сегментувати трафік через VLAN, ізолюючи користувацький і сенсорний трафік, запобігаючи несанкціонованому доступу.

Моніторинг і діагностика мережі реалізовані ПЗ на основі алгоритмів машинного навчання. Класифікаційна модель аналізує телеметричні дані, прогнозуючи ймовірність відмов мережевого обладнання. Це дозволяє проактивне обслуговування інфраструктури, виявляючи потенційні проблеми.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Строкань О., Назаров Є. Система управління «Розумний будинок» на основі технології Internet of Things // Вісник Національного технічного університету «ХПІ». Серія: Нові рішення у сучасних технологіях. 2022. №2 (12). С. 42–47.

2. Ковальова Ю.В. Моделювання топології бездротових сенсорних мереж // Системні технології. 2021. №1 (132). С. 92–99.

3. Безрук В.М., Скорик Ю.В., Нестеренко З.В. Вибір переважних протоколів маршрутизації бездротової сенсорно-актуаторної мережі методом аналізу ієрархій // Вісник Національного технічного університету «ХПІ». Серія: Нові рішення у сучасних технологіях. 2020. №2 (12). С. 30–35.

4. Чернобай К.Ю., Грибков С.В. Аналіз та шляхи вирішення проблем захисту комерційних бездротових локальних мереж Wi-Fi // Наукові праці Національного університету харчових технологій. 2017. Т. 23, №2. С. 99–106.

5. Гераїмчук М.Д. Розробка теоретичних основ побудови сенсорних мереж на основі сучасних інформаційних і нано-мікро технологій // Завершені науково-дослідні роботи КПІ ім. Ігоря Сікорського. URL: https://report.kpi.ua/uk/0108u000401 (дата звернення 06.03.2025).

6. Hadi, M. U., Song, J., Soman, S. K. O., Rahimian, A., & Cheema, A. A. Experimental Evaluation of Hybrid Fibre–Wireless System for 5G Networks. Telecom. 2022. Vol. 3 No 2, P. 218-233.

7. Perrig A., Szewczyk R., Wen V., Culler D., Tygar J.D. SPINS: Security Protocols for Sensor Networks // Wireless Networks. 2022. Vol. 8, No. 5. P. 521–534.

8. Baronti P., Pillai P., Chook V.W.C., Chessa S., Gotta A., Hu Y.F. Wireless Sensor Networks: A Survey on the State of the Art and the 802.15.4 and ZigBee Standards // Computer Communications. 2017. Vol. 30, No. 7. P. 1655–1695.

9. Layered Protocol Stack of IEEE 802.15.4 The Physical layer defined by... | Download Scientific Diagram. URL: https://www.researchgate.net/figure/Layered-

Protocol-Stack-of-IEEE-802154-The-Physical-layer-defined-by-IEEE802154-is_fig1_283042558 (дата звернення 06.03.2025).

10. Khorov E., Kiryanov A., Lyakhov A., Bianchi G. A Tutorial on IEEE 802.11ax High Efficiency WLANs // IEEE Communications Surveys & Tutorials. 2019. Vol. 21, No. 1. P. 197–216.

11. Bellalta B. IEEE 802.11ax: High-Efficiency WLANs // IEEE Wireless Communications. 2019. Vol. 23, No. 1. P. 38–46.

12. WiFi protocol stack. URL: https://www.futurelearn.com/info/courses/cybercrime-prevention-and-protection/0/steps/340104 (дата звернення 06.03.2025).

13. Application of NB-IoT Accessories in Smart Home Automation Industry // GAO Tek Inc. URL: https://gaotek.com/application-of-nb-iot-accessories-in-smart-home-automation-industry/ (дата звернення 06.03.2025).

14. NB-IoT Protocol Stack | LTE-NB Protocol Stack. URL: https://www.rfwireless-world.com/Terminology/LTE-NB-IoT-Protocol-Stack.html (дата звернення 06.03.2025).

15. Netgear's new Wi-Fi 7 Orbi mesh system is a solid bump with plenty of ports // The Verge. 2025. URL: https://www.theverge.com/2025/1/7/24337658/netgear-orbi-870-wi-fi-7-mesh-system-price-features-ces-2025 (дата звернення 06.03.2025).

16. Orbi hardwired satellite issues - can't see satell. - NETGEAR Communities. URL: https://community.netgear.com/t5/Orbi-Wi-Fi-5-AC-and-Orbi-with/Orbi-hardwired-satellite-issues-can-t-see-satellite-browser/td-p/1626375 (дата звернення 06.03.2025).

17. Networking, C., Hernández, L., Kurniawan, F., & Alfred, R.8 A Wireless Infrastructure. The Spirit of Recovery: IT Perspectives, Experiences, and Applications during the COVID-19 Pandemic. 2024. pp. 99-102.

18. Meraki Cloud Architecture - Cisco Meraki Documentation. URL: https://documentation.meraki.com/Architectures_and_Best_Practices/Cisco_Meraki_Best_Practice_Design/Meraki_Cloud_Architecture (дата звернення 06.03.2025).

19. Voicu, I. F., Diaconu, D. C., & Diaconu, D. C. Unauthorized access control in water utility computer networks. In International Conference on Machine Intelligence & Security for Smart Cities (TRUST) Proceedings. 2024. Vol. 1, pp. 79-88.

20. Ubiquiti Unifi Networking Equipment | ServersPlus. URL: https://www.serversplus.com/ubiquiti-unifi (дата звернення 06.03.2025).

21. Factory Data. URL: https://www.kaggle.com/datasets/woonel/factory-data-classification (date of access: 13.03.2025).

ДОДАТОК А
НАЛАШТУВАННЯ

Лістинг А.1 – Налаштування інтерфейсів центрального комутатора

```
interface Port-channel1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
!
interface Port-channel2
 switchport trunk allowed vlan 100-101
 switchport mode trunk
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
 channel-group 1 mode desirable
!
interface GigabitEthernet1/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
 channel-group 1 mode desirable
!
interface GigabitEthernet2/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
 channel-group 2 mode desirable
!
interface GigabitEthernet3/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
 channel-group 2 mode desirable
!
interface GigabitEthernet4/1
 switchport trunk allowed vlan 100-101
 switchport mode trunk
!
interface GigabitEthernet5/1
 switchport access vlan 100
 switchport mode access
!
interface GigabitEthernet6/1
 switchport access vlan 101
 switchport mode access
!
interface GigabitEthernet7/1
 switchport access vlan 100
 switchport mode access
!
```

```
interface GigabitEthernet8/1
 switchport access vlan 101
 switchport mode access
!
interface GigabitEthernet9/1
 switchport trunk allowed vlan 101
 switchport mode trunk
!
…
!
interface Vlan100
 no ip address
!
interface Vlan101
 no ip address
```

ДОДАТОК Б
ЛІСТИНГИ ПРОГРАМ

Лістинг Б.1 – Програма контролера MCU

```
var state = 0;
var wind = 0;
var change = true;

function setup() {
pinMode(0, INPUT);
IoEClient.setup({
   type: "Press Button",
   states: [{
       name: "Press",
       type: "bool"
    },
    {
       name: "Wind",
       type: "number"
    },
   ]
});
}

function loop() {
var st = digitalRead(0);

if (state != st) {
     state = st;
     change = true;
}

var w = analogRead(A0);
if (wind != w) {
     wind = w;
     change = true;
}

if (change) {
  IoEClient.reportStates([state?true:false, wind]);
  change = false;
}

      delay(500);
}
```

Лістинг Б.2 – Лістинги програмного коду

```
using MonitoringApp.AppCode;
using MonitoringApp.Forms.Systems;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Globalization;
using System.IO;
using System.Diagnostics;
using Microsoft.ML.Transforms;

namespace MonitoringApp.Forms.Systems {
  public partial class ModelsForm : Form {
    private MLContext mlContext;
    ITransformer trainedModel;
    private IDataView dataView;
    private string _Path = "";

    private int _selectedRowIndex = 0;
    private ValidationMy _Validation = new ValidationMy();
    private ModelsProvider _ModelsProvider = new
ModelsProvider();
    private List<Models> _ModelsList = new List<Models>();
    private LogsProvider _LogsProvider = new LogsProvider();
    private bool _IsModelTrain = false;
    private Random _rand = new Random();

    public ModelsForm() {
      InitializeComponent();
      DataLoad();
    }

    private async void OpenBtn_Click(object sender, EventArgs e)
{
      // Створення діалогового вікна для відкриття файлу
      OpenFileDialog openFileDialog = new OpenFileDialog();
      // Налаштування властивостей діалогового вікна
      openFileDialog.Filter = "Файли CSV (*.csv)|*.csv|Усі файли
(*.*)|*.*";
      openFileDialog.FilterIndex = 2;
      openFileDialog.RestoreDirectory = true;

      // Відображення діалогового вікна та обробка результату
      if (openFileDialog.ShowDialog() == DialogResult.OK) {
        _Path = openFileDialog.FileName;
        FileNameTBox.Text = openFileDialog.FileName;

        // Створення контексту ML
        mlContext = new MLContext(seed: 0);

        // Завантаження даних як RawMachineData
```

```
            dataView                                                    =
mlContext.Data.LoadFromTextFile<RawMachineData>(
                _Path,
                separatorChar: ',',
                hasHeader: true);

            // Конвертація DataView у перелік записів для підрахунку
            var dataEnumerable =

mlContext.Data.CreateEnumerable<RawMachineData>(dataView,
              reuseRowObject: false);

            // Підрахунок позитивних та негативних зразків
            int   positiveCount   =   dataEnumerable.Count(data   =>
data.Label);
            int   negativeCount   =   dataEnumerable.Count(data   =>
!data.Label);

            // Додавання інформації у текстове поле
            RaportTBox.Text += "Дані успішно завантажені\r\n";
            RaportTBox.Text += $"Назва класу: Machine Status\r\n";
            RaportTBox.Text   +=   $"Позитивні   зразки   (відмови):
{positiveCount}\r\n";
            RaportTBox.Text += $"Негативні зразки (нормальний стан):
{negativeCount}\r\n";
            Application.DoEvents();

            // Запуск асинхронного тренування моделі
            await TrainModelAsync();
          }
        }



        private async Task TrainModelAsync() {
          // Розділення даних на тренувальні та тестові
          var trainTestData =
            mlContext.Data.TrainTestSplit(dataView,    testFraction:
0.2);
          var trainingData = trainTestData.TrainSet;
          var testData = trainTestData.TestSet;

          // Побудова та тренування моделі
          var                   trainingPipeline                    =
mlContext.Transforms.ReplaceMissingValues("AmbientTemperature",
            replacementMode:
MissingValueReplacingEstimator.ReplacementMode.Mean)

.Append(mlContext.Transforms.ReplaceMissingValues("ProcessTemperatur
e",

              replacementMode:
MissingValueReplacingEstimator.ReplacementMode.Mean))
```

```
.Append(mlContext.Transforms.ReplaceMissingValues("RotationSpeed",
                replacementMode:
MissingValueReplacingEstimator.ReplacementMode.Mean))

.Append(mlContext.Transforms.ReplaceMissingValues("Torque",
                replacementMode:
MissingValueReplacingEstimator.ReplacementMode.Mean))

.Append(mlContext.Transforms.ReplaceMissingValues("ToolWear",
                replacementMode:
MissingValueReplacingEstimator.ReplacementMode.Mean))
                .Append(mlContext.Transforms.Concatenate("Features",
                    nameof(RawMachineData.AmbientTemperature),
                    nameof(RawMachineData.ProcessTemperature),
                    nameof(RawMachineData.RotationSpeed),
                    nameof(RawMachineData.Torque),
                    nameof(RawMachineData.ToolWear)))

.Append(mlContext.BinaryClassification.Trainers.FastTree(
                    numberOfLeaves: 20,
                    numberOfTrees: 100,
                    minimumExampleCountPerLeaf: 10));

            // Вимірювання часу тренування
            var stopwatch = System.Diagnostics.Stopwatch.StartNew();

            // Перевірка наявності даних для тренування
            if
(mlContext.Data.CreateEnumerable<RawMachineData>(trainingData,
              reuseRowObject: false).Any()) {
              trainedModel      =      await      Task.Run(()      =>
trainingPipeline.Fit(trainingData));
              stopwatch.Stop();

              RaportTBox.Text += $"Модель успішно натреновано за" +
                $" {stopwatch.Elapsed.TotalSeconds:N2} секунд.\r\n";

              // Оцінка моделі
              var predictions = trainedModel.Transform(testData);
              var                        metrics                        =
mlContext.BinaryClassification.Evaluate(predictions, "Label");

              // Основні метрики
              RaportTBox.Text              +=              $"Точність:
{metrics.Accuracy:P2}\r\n";
              RaportTBox.Text += $"Площа під ROC-кривою (AUC):" +
                $" {metrics.AreaUnderRocCurve:P2}\r\n";
              RaportTBox.Text += $"F1-міра: {metrics.F1Score:P2}\r\n";
```

Продовження лістингу Б.2

```
              // Метрики по кожному класу
              RaportTBox.Text += "\r\n=== Metrics per Class ===\r\n";
```

```
            RaportTBox.Text += "Позитивний клас (1):\r\n";
            RaportTBox.Text             +=             $"Precision:
{metrics.PositivePrecision:P2}\r\n";
            RaportTBox.Text               +=               $"Recall:
{metrics.PositiveRecall:P2}\r\n";
            double positiveF1Score = 2 * metrics.PositivePrecision *
metrics.PositiveRecall /
                                     (metrics.PositivePrecision   +
metrics.PositiveRecall);
            RaportTBox.Text         +=         $"F1         Score:
{positiveF1Score:P2}\r\n";

            RaportTBox.Text += "Негативний клас (0):\r\n";
            RaportTBox.Text             +=             $"Precision:
{metrics.NegativePrecision:P2}\r\n";
            RaportTBox.Text               +=               $"Recall:
{metrics.NegativeRecall:P2}\r\n";
            double negativeF1Score = 2 * metrics.NegativePrecision *
metrics.NegativeRecall /
                                     (metrics.NegativePrecision   +
metrics.NegativeRecall);
            RaportTBox.Text         +=         $"F1         Score:
{negativeF1Score:P2}\r\n";

            // Матриця помилок
            var                     scoredData                     =
mlContext.Data.CreateEnumerable<FailurePrediction>(
                predictions, reuseRowObject: false).ToList();
            int truePositives = 0;
            int trueNegatives = 0;
            int falsePositives = 0;
            int falseNegatives = 0;
            foreach (var prediction in scoredData) {
              bool actual = prediction.Label; // Реальний клас
              bool   predicted   =   prediction.PredictedLabel;   //
Передбачений клас
              if (actual && predicted)
                truePositives++;
              else if (!actual && !predicted)
                trueNegatives++;
              else if (!actual && predicted)
                falsePositives++;
              else if (actual && !predicted)
                falseNegatives++;
            }
            RaportTBox.Text += "\r\n=== Confusion Matrix ===\r\n";
            RaportTBox.Text        +=        $"True        Positives:
{truePositives}\r\n";

            RaportTBox.Text        +=        $"True        Negatives:
{trueNegatives}\r\n";
            RaportTBox.Text       +=       $"False       Positives:
{falsePositives}\r\n";
```

```
            RaportTBox.Text       +=       $"False       Negatives:
{falseNegatives}\r\n";

            _IsModelTrain = true;
          } else {
            RaportTBox.Text  +=  "Недостатньо  даних  для  тренування
моделі.\r\n";
          }
        }



        private   void   ModelsGridView_CellClick(object   sender,
DataGridViewCellEventArgs e) {
          if    (e.ColumnIndex    ==    5    &&    ModelsGridView[0,
e.RowIndex].Value.ToString() != _ModelsList[0].Message) {
            if  (MessageBox.Show("Ви  дійсно  хочете  видалити  цю
модель?", "Видалити", MessageBoxButtons.YesNo) == DialogResult.Yes) {

_ModelsProvider.DeleteModelsByModelsId(Convert.ToInt32(ModelsGridVie
w[0, e.RowIndex].Value.ToString()));
              DataLoad();
            }
          }
        }

        private void SaveBtn_Click(object sender, EventArgs e) {
          if (IsDataEnteringCorrect()) {
            //Зберігання моделі
            string  pathName  =  @"\teach\"  +  GenerateFileName()  +
".zip";
            string localProj =

System.IO.Path.GetDirectoryName(System.Reflection.Assembly.GetExecut
ingAssembly().Location);
            _ModelsProvider.InsertModels(ModelsNamesTBox.Text,
pathName);
            mlContext.Model.Save(trainedModel,      dataView.Schema,
localProj + pathName);
            ClearAllData();
            _LogsProvider.InsertLogs(LoginForm.CurrentUser.UsersId,
              "Було навчено модель " +
              ModelsNamesTBox.Text, DateTime.Now);
            MessageBox.Show("Дані успішно збережено!");
          }
        }

        private void ClearBtn_Click(object sender, EventArgs e) {
          ClearAllData();
        }

        private void ExitBtn_Click(object sender, EventArgs e) {
          this.Close();
```

```
        }


        public string GenerateFileName() {
          DateTime now = DateTime.Now;
          string fileName = string.Format("{0}_{1}_{2}_{3}_{4}_{5}",
              now.Year, now.Month, now.Day, now.Hour, now.Minute,
now.Second);

          return fileName;
        }

        private void ClearAllData() {
          _IsModelTrain = false;
          ModelsNamesTBox.Text = String.Empty;
          RaportTBox.Text = String.Empty;
          DataLoad();
        }
        private bool IsDataEnteringCorrect() {
          bool isCorrect = true;
          if (!_IsModelTrain) {
            MessageBox.Show("Неможливо зберегти дані. \r\nЩе не
навчено модель!", "Увага!");
            isCorrect = false;
          }
          if (_Validation.IsDataEntering(ModelsNamesTBox.Text)) {
            ModelsNamesValidationLbl.Text                          =
NamesMy.ProgramButtons.RequiredValidation;
          } else {
            ModelsNamesValidationLbl.Text                          =
NamesMy.ProgramButtons.ErrorValidation;
            isCorrect = false;
          }
          return isCorrect;
        }

        private void DataLoad() {
          int firstRowIndex = 0;
          if (ModelsGridView.FirstDisplayedScrollingRowIndex > 0) {
            firstRowIndex                                          =
ModelsGridView.FirstDisplayedScrollingRowIndex;
          }
          try {
            _ModelsList = _ModelsProvider.GetAllModels();
            LoadDataInModelsGridView(_ModelsList);
            if (_selectedRowIndex == ModelsGridView.Rows.Count) {
              _selectedRowIndex = ModelsGridView.Rows.Count - 1;
            }
            if (_selectedRowIndex >= 0) {
              ModelsGridView.FirstDisplayedScrollingRowIndex       =
firstRowIndex;
              ModelsGridView.Rows[_selectedRowIndex].Selected      =
true;
```

```
            }
          } catch (Exception ex) {
            MessageBox.Show(ex.ToString());
          }
        }


    public class FailurePrediction {
      public bool PredictedLabel { get; set; } // Передбачений клас
      public float Probability { get; set; }   // Ймовірність передбачення
      public float Score { get; set; }         // Оцінка передбачення
      public bool Label { get; set; }          // Реальний клас
    }
```

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК І ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖІ

Проєктування мережі передачі даних котеджного комплексу із застосуванням стандартів сенсорних мереж

Презентація дипломного пректу

Виконав:
ПІТЮРЕНКО Нікіта Костянтинович
студент групи КНТ-512сп

Керівник:
КИРИЧЕК Галина Григорівна
к.т.н., доцент кафедри
комп'ютерних систем та мереж

2025

МЕТА ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи

Розробка макету мережевої інфраструктури котеджного комплексу, моделювання та налаштування бездротового покриття, а також реалізація програмного забезпечення для аналізу та прогнозування стану мережі.

Предмет дослідження

Методи проєктування, налаштування та інтелектуального моніторингу мережевої інфраструктури котеджного комплексу з використанням сенсорних технологій.

Об'єкт розробки

Проєкт мережі передачі даних для котеджного комплексу з інтеграцією сенсорних технологій та системи прогнозування стану мережі.

АКТУАЛЬНІСТЬ ТА АНАЛІЗ СУЧАСНИХ ПІДХОДІВ

- Розвиток IT та впровадження концепцій «Розумного будинку» й IoT підвищують вимоги до мережевої інфраструктури житлових комплексів. Виникає потреба в надійних, масштабованих мережах для з'єднання численних пристроїв на великих територіях. Сенсорні мережі є оптимальним рішенням завдяки гнучкості та енергоефективності, але їх проєктування та моніторинг є актуальними викликами.
- **Технології побудови мереж:**

– оптоволоконні мережі (FTTH/FTTB);

– бездротові технології (Wi-Fi 802.11ax, LTE/5G);

– гібридні рішення.

- Бездротові технології простіші в розгортанні, але можуть мати проблеми з покриттям великих територій. Гібридні рішення поєднують переваги обох підходів для досягнення стабільності та мобільності.

3

СТАНДАРТИ СЕНСОРНИХ МЕРЕЖ ТА ЇХ ЗАСТОСУВАННЯ

- Сенсорні мережі забезпечують автоматизований контроль за мікрокліматом, доступом, відеоспостереженням та освітленням. Вони працюють у складі єдиної цифрової екосистеми з централізованим управлінням, використовуючи для зв'язку такі протоколи, як ZigBee, LoRaWAN та NB-IoT.
- **Ключові аспекти сенсорних мереж:**

– автоматизований контроль;

– інтеграція в єдину екосистему;

– використання спеціалізованих бездротових протоколів.

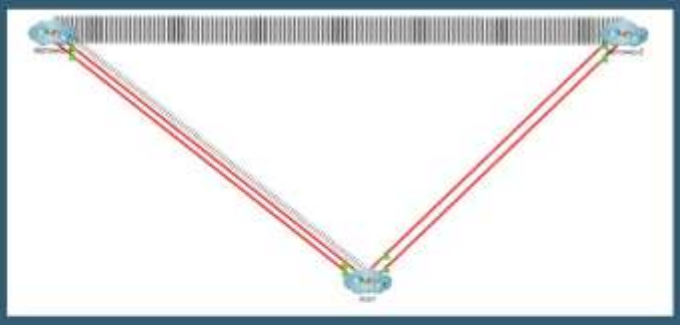
4

ТЕХНІЧНІ ВИМОГИ

- Створення надійної, масштабованої та безпечної інфраструктури, здатної підтримувати стабільне дротове та бездротове підключення користувачів і IoT-пристроїв, з централізованим управлінням, ефективним розподілом ресурсів та можливістю автоматизованого моніторингу.
  - фізична інфраструктура;
  - мережеве розгалуження;
  - бездротовий доступ;
  - сенсорна мережа;
  - система сповіщення;
  - захист інформації;
  - моніторинг та діагностика.
- Ці вимоги включають наявність центрального сервера IoT та WLC-контролерів, розгалуження на два кластери, підключені оптоволокном, використання Wi-Fi 6 з VLAN, підтримку сенсорів та систем сповіщення, захист за допомогою NAT та ACL, а також наявність програмного забезпечення для моніторингу.

5

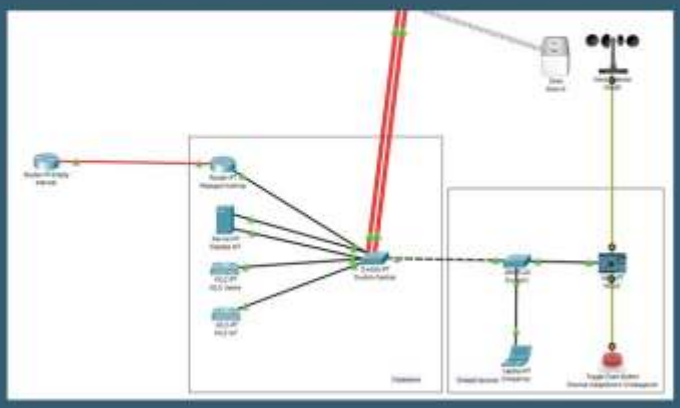
ЗАГАЛЬНА СХЕМА МЕРЕЖІ КОТЕДЖНОГО КОМПЛЕКСУ



- Мережа має трирівневу структуру, що включає центральний вузол «Хост» і два віддалені вузли – «Містечко-1» та «Містечко-2». З'єднання реалізовано через оптоволоконні канали з резервуванням. У «Хості» розміщено серверну частину, а в «Містечках» — житлові будинки з IoT-пристроями.
- Архітектура:
  - трирівнева структура;
  - оптоволоконні канали зв'язку;
  - центральний вузол «Хост»;
  - кластери «Містечка» з IoT-пристроями;
  - резервування каналів.

6

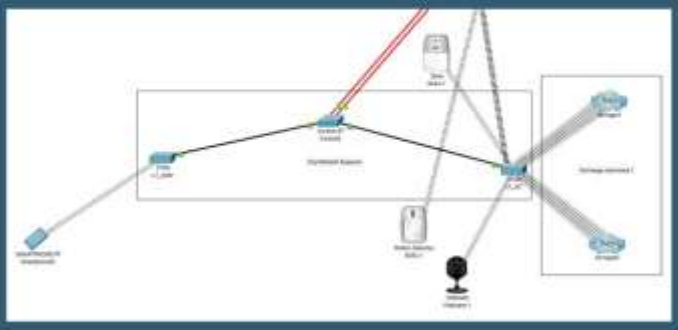
СХЕМА ХОСТУ (СЕРВЕРНОЇ)



- Серверна включає центральний комутатор для розподілу трафіку, маршрутизатор (з DHCP-сервером), сервер IoT для керування сенсорами та два WLC-контролери для користувацької та сенсорної мереж. В операторській розташовано робоче місце та пристрої аварійного сповіщення.
- **Основні компоненти:**
  - центральний комутатор (Switch-Central);
  - головний маршрутизатор (Router-PT);
  - сервер IoT (Server-PT);
  - WLC-контролери;
  - операторська;
  - доступ до Інтернету через NAT.

7

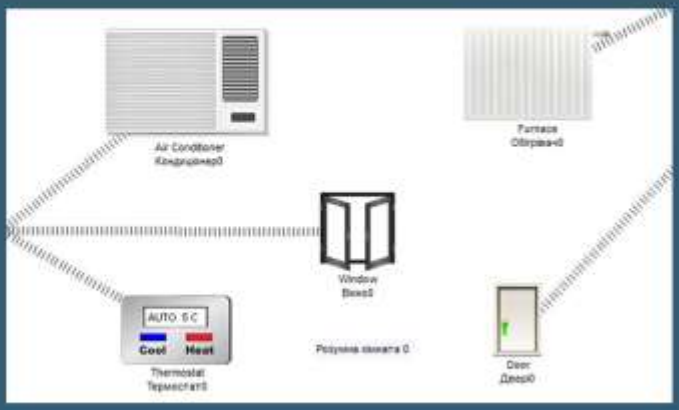
СХЕМА МІСТЕЧКА



- Центральним елементом є комутатор, підключений до хосту через гіперканал. Дві точки доступу, L1_User та L1_IoT, забезпечують ізольований трафік для користувачів та IoT-пристроїв відповідно. До мережі інтегровані сирена, детектор руху та відеокамера.
- Інфраструктура містечка:
  - мережевий комутатор;
  - бездротові точки доступу (`L1_User`, `L1_IoT`);
  - підключення котеджів;
  - IoT-пристрої безпеки та моніторингу.

8

СХЕМА КОТЕДЖУ



- Система включає термостат для регулювання температури, який керує кондиціонером та обігрівачем. Датчики на вікнах та дверях інтегровані в систему для безпеки та енергоефективності. Управління здійснюється централізовано через сервер IoT.
- **IoT-пристрої для керування мікрокліматом та безпекою:**
  - термостат;
  - кондиціонер;
  - обігрівач;
  - датчики на вікні та дверях;
  - централізоване керування.

9

ПЛАНУВАННЯ БЕЗДРОТОВОГО ДОСТУПУ ТА VLAN

- **Два типи бездротового доступу:**
  - Для звичайних користувачів.
  - Для IoT-пристроїв.
- **Сегментація за допомогою VLAN:**
  - оптимізація маршрутизації;
  - запобігання перевантаженню;
  - ізоляція критичних IoT-пристроїв.
- Сегментація реалізується на рівні комутаторів для оптимізації трафіку та ізоляції IoT-сегмента від користувацького.

VLAN	Тип	Адресація
100	Користувачі	192.168.0.0/24
101	IoT-прилади	192.168.1.0/24

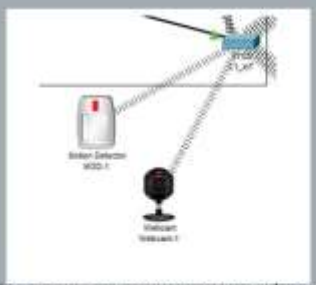
- Роль DHCP-сервера виконує маршрутизатор, DNS розташований на сервері IoT, а керування бездротовою мережею здійснюється через окремі WLC-контролери для користувачів та IoT-пристроїв.

10

НАЛАШТУВАННЯ СЕНСОРНОЇ МЕРЕЖІ ТА ЗАПИС ПОДІЙ

- **Реєстрація IoT-пристроїв на сервері:**
  - централізоване керування;
  - синхронізація даних;
  - віддалене керування.
- Підключення до єдиного сервера IoT забезпечує централізоване керування та синхронізацію даних.

Налаштування підключення до серверу IoT



Спрацьовування детектору руху та ввімкнення камери

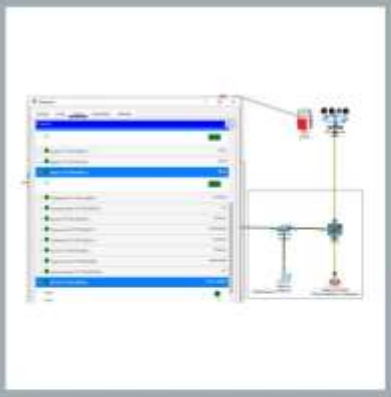
Умова для ввімкнення камери на Містечко-1

НАЛАШТУВАННЯ СЕНСОРНОЇ МЕРЕЖІ ТА ЗАПИС ПОДІЙ

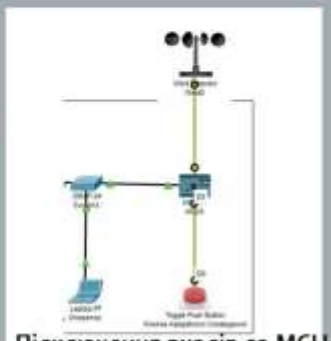
- **IoT-монітор оператора:**
  - Відображення списку активних пристроїв (термостати, кондиціонери, сирени, детектори руху тощо).
  - Контроль роботи в реальному часі.
- Інтерфейс дозволяє оператору відстежувати стан усіх пристроїв.
- **Автоматизація відеоспостереження (приклад Містечко-1):**
  - Правило «Motion-town-1»: Якщо детектор руху MOD-1 активний (On is true), тоді ввімкнути камеру Webcam-1 (On to true).
  - Правило «Motion-town-1-off»: Якщо детектор руху MOD-1 неактивний (On is false), тоді вимкнути камеру Webcam-1 (On to false).

НАЛАШТУВАННЯ СИСТЕМИ АВАРІЙНОГО СПОВІЩЕННЯ

- **Призначення полягає в** оперативному інформуванні про надзвичайні ситуації.
- Принцип роботи:
  - Тригери: Натискання кнопки тривоги або перевищення швидкості вітру.
  - Компоненти: Контролер MCU, сирени, датчики. Підключення через VLAN 101.
  - Логіка: Контролер MCU збирає дані з датчиків і активує сирени.
- **Тестування, результат.** Система успішно протестована шляхом моделювання натискання кнопки тривоги та підвищення швидкості вітру. Підтверджено одночасну активацію сповіщення на всіх майданчиках.



Аварійне сповіщення ввімкнене

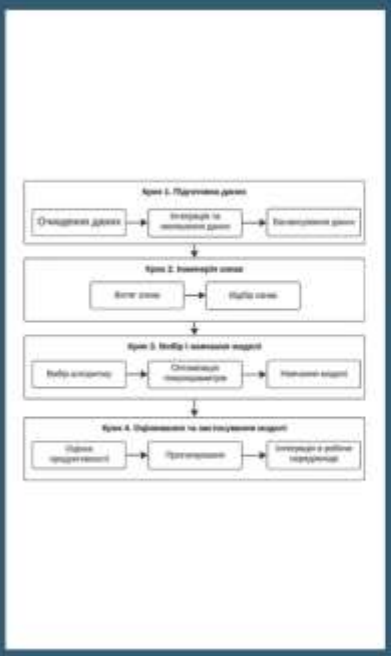


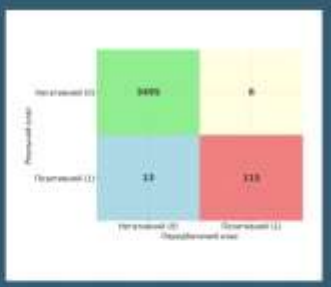
Підключення входів до MCU

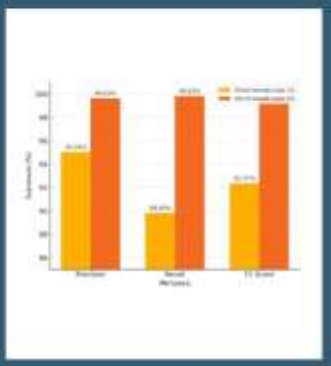
13

МАШИННЕ НАВЧАННЯ ДЛЯ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ СТАНУ МЕРЕЖІ

- **Мета:** Забезпечення надійного аналізу та прогнозування стану мережевої інфраструктури, виявлення потенційних несправностей.
- **Обраний алгоритм:** Швидке дерево рішень (FastTree) – висока швидкість навчання та стійкість до шуму.
- Було завантажено дані, що містять 624 зразки відмов та 17821 зразок нормального стану. Модель було успішно натреновано з точністю 99,48% та F1-мірою 92,37%.







14

ВИСНОВКИ

- **Розроблено проєкт мережі:** Створено комплексну мережу передачі даних для котеджного комплексу, що інтегрує сенсорні технології, забезпечуючи надійний зв'язок та централізоване управління.
- **Обрано технології:** Для бездротового доступу використано Wi-Fi 6, для сенсорного сегменту – енергоефективні NB-IoT та ZigBee (як загальний підхід, хоча в записці основний акцент на Wi-Fi для IoT).
- **Впроваджено гнучкість та безпеку:** Динамічна IP-адресація (DHCP), захист трафіку за допомогою NAT, контроль доступу через ACL, сегментація трафіку за допомогою VLAN (802.1Q).
- **Реалізовано моніторинг та діагностику:** Розроблено програмне забезпечення на основі алгоритмів машинного навчання (швидке дерево рішень) для аналізу телеметричних даних та прогнозування ймовірності відмов мережевого обладнання.
- **Підтверджено працездатність:** Тестування підтвердило коректність налаштувань, стабільну роботу всіх компонентів, розподіл IP-адрес та доступність мережевих сервісів.
- **Практичне значення:** Проєкт забезпечує ефективну, безпечну та масштабовану систему передачі даних, що відповідає сучасним вимогам до інтелектуальних житлових середовищ та дозволяє проактивне обслуговування інфраструктури.

15

ПУБЛІКАЦІЇ

- 1. Пітюренко Н.К., Киричек Г.Г. Мережа передачі даних із застосуванням стандартів сенсорних мереж. Тиждень науки-2025. Факультет комп'ютерних наук і технологій. Тези доповідей науково-практичної конференції, Запоріжжя, 14-18 квітня 2025 р. – Запоріжжя: НУ «Запорізька політехніка», 2025

16