

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
« ____ » _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи проектування архітектури
Wi-Fi мереж для операторів телекомунікаційних послуг”

КБГЗ - 2025

Виконав здобувач вищої освіти
IV курсу, групи КІ-21-2
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Ховрящук О.О.
« ____ » _____ 2025 р.

Керівник проекту
кандидат фізико-математичних наук, доцент
_____ Петренюк В.І.
« ____ » _____ 2025 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітній ступінь бакалавр
Галузь знань . 12 “Інформаційні технології”
Спеціальність 123 “Комп’ютерна інженерія”
Освітньо-професійна (освітньо-наукова) програма “Комп’ютерна інженерія”

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 17 » січня 2025 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ПЕРШИМ (БАКАЛАВРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Ховрящуку Олександр Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг

2. Керівник роботи Петренюк Володимир Ілліч, канд. фіз.-мат. наук, доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 47-02 від 17.01.2025 року

3. Строк подання студентом роботи до захисту 23.05.2025 р.

4. Мета та завдання випускної кваліфікаційної роботи: Метою роботи є розробка програмного забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Етапи програмування системи.

5. Впровадження системи в промислову експлуатацію.

6. Висновки

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

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

7. Дата видачі завдання « 17 » січня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти | Строк виконання етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти | Примітка |
|-------|---|---|----------|
| 1. | Аналіз існуючих систем | 10.03.2025 р. | |
| 2. | Постановка задачі, оформлення ТЗ | 15.03.2025 р. | |
| 3. | Розробка моделі компонента | 20.03.2025 р. | |
| 4. | Розробка структур даних | 25.03.2025 р. | |
| 5. | Розробка алгоритмів зв'язку та відображення | 30.03.2025 р. | |
| 6. | Програмування алгоритмів | 10.04.2025 р. | |
| 7. | Оформлення ПЗ | 17.04.2025 р. | |
| 8. | Попередній захист роботи | 23.05.2025 р. | |
| | | | |
| | | | |
| | | | |
| | | | |

Дата видачі завдання
« 17 » січня 2025 р.

Підпис керівника

Петренюк В.І.
(прізвище та ініціали)

Завдання прийнято до виконання
« 17 » січня 2025 р.

Підпис здобувача

Ховрящук О.О.
(прізвище та ініціали)

АНОТАЦІЯ

Ховрящук О.О. Програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Метою розробки є програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Результат роботи – програмна реалізація системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, проектування архітектури Wi-Fi мереж

ABSTRACT

Khovryaschuk O.O. Software for the Wi-Fi network architecture design system for telecommunication service operators. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed, which is intended for the Wi-Fi network architecture design system for telecommunication service operators.

The purpose of the development is software for the Wi-Fi network architecture design system for telecommunication service operators.

The result of the work is the software implementation of the Wi-Fi network architecture design system for telecommunication service operators.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software are provided.

The program can be used on PCs with Windows 10/11.

The program is developed in Python.

Keywords: computer engineering, Wi-Fi network architecture design

ЗМІСТ

| | |
|---|----|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ | 2 |
| ВСТУП..... | 3 |
| 1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ | 5 |
| 1.1 Призначення системи..... | 5 |
| 1.2 Область застосування..... | 7 |
| 2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ | 8 |
| 2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти..... | 8 |
| 2.2 Обґрунтування вибору засобів для побудови системи та мови програмування..... | 12 |
| 2.3 Розгорнута постановка завдання | 14 |
| 3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ | 16 |
| 3.1 Опис функціонування системи | 16 |
| 3.2 Розробка структурної схеми..... | 20 |
| 3.3 Розробка функціональної схеми | 32 |
| 3.4 Розробка діаграми процесів..... | 44 |
| 4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ..... | 46 |
| 4.1 Розробка блок-схем та опис алгоритмів функціонування системи..... | 46 |
| 4.2 Захист розробленого програмного забезпечення..... | 58 |
| 5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ | 61 |
| 6 ОСНОВНІ ВИСНОВКИ..... | 68 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ | 70 |

| | | | | | | | | |
|----------|----------------|----------|-------|------|--|---------------------------|-------|---------|
| | | | | | | ВКРБ-123.25.0044.00.00.ПЗ | | |
| Вим. | Арк. | № докум. | Підп. | Дата | | | | |
| Розроб. | Ховряцук О.О. | | | | Програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг | Літ. | Аркуш | Аркушів |
| Перев. | Петренко В.І. | | | | | Б | 1 | 76 |
| Н.контр. | Коваленко А.С. | | | | ЦНТУ КІ-21-2 | | | |
| Затв. | Смірнов О.А. | | | | | | | |

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

| | | |
|------|---|---|
| БНМ | – | балансування навантаження мережі, Network Load Balancing |
| ЕЦП | – | електронний цифровий підпис |
| ІВК | – | інфраструктура відкритих ключів |
| ІТ | – | інформаційні технології |
| ЛОМ | – | локальна обчислювальна мережа |
| ПЗ | – | програмне забезпечення |
| СКЗІ | – | система комплексного захисту інформації |
| УК | – | управляючі компоненти |
| УЦ | – | удостовірюючий центр |
| IGMP | – | Internet Group Management Protocol |
| NLB | – | Network Load Balancing, балансування мережного навантаження |
| PKI | – | Public Key Infrastructure |
| VPN | – | віртуальна приватна мережа |

КБПЗ-2025

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 2 |

ВСТУП

Актуальність теми. Колись планування мережі Wi-Fi передбачало створення плану поверху та нанесення точок доступу за допомогою транспортира. Це була виснажлива та трудомістка діяльність. Точки доступу більше не потребують ручного налаштування живлення та каналів. Планування бездротового зв'язку також стало набагато простішим завдяки чудовим можливостям програмного забезпечення. Однак недоліком є те, що бездротові пристрої та світ Wi-Fi продовжують ставати складнішими, ніж будь-коли.

Спочатку розробка мережі Wi-Fi передбачала великий наголос на охопленні, оскільки лише вибрана група людей використовувала та покладалася на бездротовий зв'язок на регулярній основі. Оскільки все більше компаній обирають BYOD, важко знайти звичайного користувача, який би не поклався на бездротовий зв'язок щодня та часто на кількох пристроях. Зайве говорити, що дизайн бездротової мережі значно змінився.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.
- Дослідження системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.
- Програмна реалізація системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 3 |

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

КБПЗ_2025

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 4 |

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Wi-Fi стає все популярніше як технологія для голосового зв'язку, альтернативна класичним стільниковим системам. Тому одним з важливих вимог до сучасних рішень Wi-Fi, поряд з високою продуктивністю і ємністю, є безшовний хендовер (стандарт IEEE 802.11r) – швидке перемикання клієнта на іншу точку доступу без перерви сервісу.

Крім того, замовники зацікавлені в тому, щоб при розгортанні мереж Wi-Fi «поменше копати й тягти кабелі». Задовольнити ця вимога, як вважає фахівець Cambium Networks, допомагає підтримка ніздрюватих топологій (mesh), а також наявність доступних за ціною підключень для точок доступу (backhaul). Зв'язок із вищестоящими вузлами може бути забезпечена, зокрема, за рахунок ресурсів самої ТД – наприклад, у діапазоні 5 ГГц. Для спрощення й прискорення розгортання мережі також важлива наявність засобів автоконфігурування й шаблонів для налаштування.

Аналізуючи наявні на ринку пропозиції виділяємо дві групи рішень:

– До першого відносяться недорогі продукти, пропоновані такими компаніями, як Ubiquiti Networks і MikroTik. Ці рішення мають надзвичайно привабливу ціну (100-300 доларів), але при цьому вони часто характеризуються поганою масштабованістю, обмеженими можливостями керування, недостатньою надійністю й невисокою ємністю. Будуються подібні мережі, як правило, на основі програмного контролера або взагалі без нього.

– Інша група – рішення з відмінними характеристиками й широкою функціональністю, пропоновані такими компаніями, як Cisco, Aruba (тепер у складі HP), Ruckus (у складі Brocade), Meru (у складі Fortinet), Zebra Technologies і поруч інших. Мережі на базі продуктів зазначених компаній часто будуються з

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 5 |

використанням апаратних контролерів. Їх відрізняють гарна радіочастотна частина, широка функціональність (у тому числі наявність засобів аналітики), керованість провайдерського рівня, більша ємність, надійність. Але й вартість таких продуктів досить висока – за оцінкою фахівця Cambium Networks, від 500 доларів, хоча слід зазначити, що в портфелі пропозицій названих вендорів всі частіше з'являються й рішення нижнього цінового діапазону.

Вийшовши на ринок Wi-Fi пізніше більшості конкурентів, компанія Cambium Networks, як затверджують її представники, вирішила сполучити переваги двох груп наявних рішень (невисока вартість першої й функціональність другої) у продуктах cnPilot. У це сімейство входять трохи ТД із підтримкою 802.11ac і 802.11n, у тому числі з можливістю одночасної роботи на частотах 2,4 і 5 Ггц. Контролер cnMaestro (по суті, це система керування) є програмним і може бути розгорнуто як локально, так і в хмарі. Останній варіант, вважають в Cambium Networks, оптимальний для малих мереж.

Архітектурна гнучкість відрізняє пропозиції практично всіх провідних постачальників рішень Wi-Fi. Так, Ruckus Wireless пропонує три варіанти архітектури:

– Перший – без виділеного контролера: у цьому випадку функції контролера реалізовані в ПЗ самих точок доступу, таке рішення забезпечує роботу в одній мережі до 25 ТД.

– Другий варіант – з контролером, що може бути виконаний як окремий пристрій або розгорнуто у приватній хмарі у вигляді ПЗ (віртуальний контролер). При цьому віртуальний і фізичний контролери мають однакову функціональність.

– Третій варіант – керування точками доступу з публічної хмари. Майже всі точки доступу, що випускаються компанією можуть використовуватися в кожній із трьох архітектур.

Коли потрібна висока щільність

Мова в цьому випадку йде не тільки про побудову бездротових мереж на стадіонах, але й про охоплення ними готелів, вокзалів, транспорту, центральних

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 6 |

вулиць та ін. У зазначених місцях будуть потрібні рішення, які одержали назву «Wi-Fi високої щільності (HD)». Саме впровадження систем високої щільності є, з погляду українських фахівців, самим актуальним напрямком розвитку бездротових мереж передачі даних.

1.2 Область застосування

Провідні українські оператори готові налаштовувати хот-споти для своїх клієнтів таким чином, щоб забезпечити споживачеві послуг:

- простоту online керування порталом без навичок програмування (персоналізація й сторінки доступу й застосування маркетингових інструментів лояльності);
- можливість автоматичного запуску привітальної сторінки в хот-спот-зоні;
- великий вибір зручних убудованих додатків для надання додаткових сервісів своїм відвідувачам;
- доступ до набору типових шаблонів для різних видів діяльності.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 7 |

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти

Wi-Fi в громадському транспорті

Доступ до швидкісного інтернету сьогодні є важливою складовою повсякденного життя багатьох українців. І пріоритетне завдання оператора «Інтертелеком» – це впровадження нових технологій з метою подолання цифрової нерівності. Ми прагнемо до того, щоб кожен житель України міг використовувати швидкісний інтернет там, де він необхідний.

Першими кроками в цьому напрямку стала організація Wi-Fi хот спотів в місцях відпочинку людей по всій країні. На сьогоднішній день в Україні працює понад 700 громадських точок доступу.

У 2013 проект отримав новий розвиток – компанія Інтертелеком розпочала реалізацію проекту «Безкоштовний Wi-Fi в громадському транспорті», забезпечивши на сьогоднішній день пасажиром більше 730 транспортних одиниць по всій країні.

У 2022 році компанія «Інтертелеком» стала партнером «Української Залізничної Швидкісної Компанії» в частині надання безкоштовного Wi-Fi на поїздах класу Інтерсіті та Інтерсіті+.

Wi-Fi в поїздах Інтерсіті+

Українська Залізнична Швидкісна компанія» і національний 3G оператор «Інтертелеком» представляють унікальний соціально-значимий для України проект з оснащення швидкісних міжміських поїздів ІНТЕРСІТІ+, інтернет-доступом за технологією CDMA з роздачею по Wi-Fi.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 8 |

Доступ в інтернет надається – **безкоштовно**, без обмеження по трафіку і часу!

Тепер пасажирів в дорозі можуть проводити час з користю. Спілкування в соціальних, мережах, музика, відео або книги онлайн доступні без обмежень.

Як активувати Wi-Fi доступ

Послугу Wi-Fi доступу в інтернет від Інтертелеком в поїздах ІНТЕРСІТІ + можна використовувати на будь-якому пристрої з Wi-Fi-адаптером: ноутбук, планшет, смартфон, ігрова приставка і т.д.

1. Активуйте Wi-Fi на пристрої, на якому плануєте використовувати інтернет-доступ.

2. Виберіть безпроводову мережу «Інтертелеком Free». Для користувачів пристроїв Apple також може відображатись мережа «Інтертелеком 5G» (роботі в частоті 5 Ггц.). Для використання можна вибрати будь-яку мережу.

3. Запустіть браузер і введіть адресу будь-якого сайту. Після цього система автоматично перенаправить вас на головну сторінку активації сервісу «Wi-Fi в Інтерсіті».

Послуга доступна будь-яким користувачам, незалежно від оператора зв'язку, який використовується в пристрої.

Установка Wi-Fi хот-спот на Android

Цього разу пропонуємо вам познайомитися з функцією, що називається портативний Wi-Fi хот-спот у пристроях Android.

Розберемо, що це таке, де дана функція буде корисна, і як їй користуватися.

Отже, що таке портативний Wi-Fi хот-спот на Android? Саме словосполучення hot spot з перекладу з англійського означає «гаряча точка».

І позначає воно певну місцевість, у якій за допомогою якогось портативного гаджета (смартфона, КПК або ноутбука), що має в собі модуль Wi-Fi, можна роздобути доступ до бездротового Інтернету або просто мережі.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 9 |

Простіше говорячи, це функція, за допомогою якої, наприклад, ваш телефон на Android може виступати в ролі точки доступу Wi-Fi, тобто роздавати Інтернет або ж просто мережа для інших портативних девайсів.

Налаштування хот-спота

Перед тим, як використовувати портативний Wi-Fi хот-спот, його потрібно налаштувати, і ми покажемо, як відбувається установка портативної Wi-Fi точки доступу Android.

Сама процедура не є складною й виконується через налаштування ОС Android.

Для цього:

- Зайдіть у налаштування апарата.
- Перейдіть по пункті «Загальний модем і мережі».
- Виберіть «Точка доступу Wi-Fi».
- Після чого потрібно буде спочатку налаштувати роздачу своєї мережі.
- Виберіть «Створити точку доступу Wi-Fi».
- У новому вікні в поле «Назву» придумайте й напишіть ім'я вашого підключення (саме воно буде відображатися на пристроях, які будуть підключатися).
- У пункті «Безпеку» виберіть значення «Відкрити», якщо хочете, щоб ваша роздача була без пароля, і до неї могли підключатися всі бажаючі, або ж WPA2 PSK, якщо збираєтеся встановити пароль.
- Примітка: WPA PSK менш надійна, чим WPA2 PSK, тому рекомендується вибирати саме останній.
- Якщо потрібно, задайте в третім полі пароль для вашої мережі, він повинен мати в собі мінімум 8 символів.
- Натисніть «Зберегти».
- Після цих дій налаштування зроблене.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 10 |

Хочемо помітити, що інші пристрої зможуть одержувати Інтернет через вашу роздачу, якщо у вас на смартфоні активний мобільний Інтернет, якщо ж ні, тоді ви будете тільки роздавати мережа.

Такий варіант теж може бути корисний, коли ви хочете роздрукувати файл зі смартфона через принтер, що оснащений Wi-Fi.

Після налаштування для включення точки доступу потрібно праворуч зверху нажати на перемикач, щоб мережа стала лунати.

Тепер залишиться тільки підключитися до неї з інших пристроїв.

Додатки для роботи

Існують безкоштовні додатки для хот-спота Wi-Fi Android, які спрощують процедуру керування хот-спот на Android.

Вони займають дуже мало місця й невимогливі до ресурсів, тому повинні працювати абсолютно на всіх апаратах.

Портативний Wi-Fi хот-спот

Перша програма називається «Портативний Wi-Fi хот-спот» і її можна скачати в Google Play безкоштовно для Android.

Функціонал додатка полягає в тому, що запустити бездротову мережу можна всього в 1 клік, з ярлика на робочому столі, і, якщо ви часто користуєтеся цією функцією, дане додаток для вас буде дуже до речі.

WiFi HotSpot

Другий Android-додаток для створення точки доступу Wi-Fi називається WiFi HotSpot, воно має такого ж функціонала, що й попередня програма, але відображається не ярликом, а як інтерактивний віджет хот-спота для Android, що також виконує запуск мережі з робочого стола.

Зміст віджета полягає в тому, що він має кольоровий індикатор, по якому можна побачити стан вашого підключення.

Якщо індикатор червоного кольору, виходить, мережа неактивна, коли мигає блакитним і зеленим – відбувається процедура включення, а якщо просто блакитного, це означає, що роздача вже працює.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 11 |

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – високорівнева мова програмування, яку називають другою за популярністю в світі. Її використовують для розробки вебзастосунків, програмного забезпечення, машинного навчання. Python застосовують для вирішення робочих завдань у компаніях Google, Instagram, Facebook, IBM, NASA, Dropbox, Netflix та інших. Розробники цінують цю мову програмування за простоту у вивченні, ефективність та мультиплатформність.

Python – скриптова мова програмування з досить простим синтаксисом. Для розуміння достатньо порівняти принципи написання найпростішої програми, яка виводить на екран текстове повідомлення. Саме тому мова програмування Python більш доступна для новачків, а професіонали встигли адаптувати її для вирішення великої кількості завдань. Це мультиплатформне рішення, тому знання Python дає можливість працювати у різних сферах: від розробки мобільних застосунків до ігрової індустрії та штучного інтелекту.

У мови програмування динамічна типізація: є можливість передавати до функцій будь-який тип даних без попереднього вказання. Інтерпретованість дозволяє знаходити помилки у коді ще до повної збірки у робочий застосунок. При цьому Python дуже чітко дає зрозуміти, де та через що виникла помилка.

Це мова об'єктноорієнтованого програмування (ООП). Програмне забезпечення на Python оформлене у вигляді моделей, які можуть бути зібраними у пакети. Тип та структуру кожного об'єкта можна запитати під час виконання програми. Для кожного з об'єктів можна отримати всю інформацію щодо його внутрішньої структури. Окрім того:

- у мови логічний синтаксис, завдяки чому вихідний код легко читати та розуміти;
- гнучкість та масштабованість Python дозволяє адаптувати високорівневу логіку та розширяти складні застосунки, як тільки виникне така необхідність;

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 12 |

- розробка на Python у більшості випадків проходить швидше, ніж на інших мовах програмування;
- Python – інтерпретована мова програмування. Це значить, що код можна написати у будь-якому текстовому файлі на будь-якій платформі, і потім успішно запустити;
- у Python – колосальна спільнота однодумців. Тож будь-які складнощі конкретних розробників вирішуються колективно.

Проте є декілька особливостей, які можна віднести до недоліків. Це повільність (ця мова програмування хоч і універсальна, проте повільніша за інші), велика кількість ресурсів, необхідних для роботи та «прив’язаність» до системних бібліотек.

Мова програмування Python використовується у наступних сферах:

1. Розробка програмних застосунків будь-якого напрямку.
2. Розробка серверної частини мобільних застосунків (найпопулярніший напрямок).
3. Ігри. Багато сучасних ігор для комп’ютерів (наприклад, World of Tanks) частково чи повністю написані на Python.
4. Вбудовані системи для різних пристроїв. Дуже часто Python використовують для написання внутрішніх платформ управління банкоматами.
5. Скрипти та плагіни до уже реалізованих програм для автоматизації процесів чи створення інших рішень.
6. Тестування (автоматизація цього процесу).
7. Машинне навчання. – основна мова для написання алгоритмів і аналітичних застосунків у сфері Machine Learning.

Бібліотеки Python

Різні бібліотеки Python використовують для виконання конкретних завдань. Наприклад, Matplotlib підходить для відображення даних у двовимірній та тривимірній графіці. Pandas підходить для зручної роботи з даними. NumPy дозволяє створювати масиви та керувати ними. Requests використовується для

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|-----------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 13 |

веброзробки. OpenCV-Python відкриває можливості для обробки зображень з метою оптимізації систем «машинного зору».

Найвідоміші фреймворки для мови програмування Python

Фреймворки Python допомагають створити зручне та функціональне середовище для розробки. У них міститься набір інструментів, модулів та бібліотек, корисних для виконання конкретних завдань. Це значно полегшує роботу: наприклад, дає змогу не витратити час на розписування дій, які повторюються, а використати релевантний інструмент. Тож є можливість позбутися рутинних процесів та сконцентруватися на логіці проекту.

Серед найпопулярніших фреймворків для Python:

– Django – найстаріший та найвідоміший. Створений для реалізації великих інтерактивних проєктів;

– Pyramid – зручний у налаштуваннях, і дає можливість реалізувати складні нестандартні ідеї;

– Web2py – підходить в першу чергу для вебзастосунків і може використовуватись на будь-яких архітектурах.

Популярні Python IDE

IDE або інтегровані середовища розробки – це програмне забезпечення, яке надає розробникам необхідні інструменти для написання, редагування, тестування та налаштування коду. Для розробки на Python найчастіше використовують IDE PyCharm, IDLE, Spyder та Atom.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

В процесі розробки випускної кваліфікаційної роботи за першим

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 14 |

(бакалаврським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 15 |

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Без належного планування мережеві команди можуть ненавмисно обійти основні організаційні вимоги, спричинити неефективні процеси та витратити цінні гроші та час.

Що таке планування бездротової мережі?

Планування бездротової мережі – це процес розробки стратегії та проектування бездротової мережі, яка забезпечує адекватне покриття, безпеку та пропускну здатність. У цьому процесі мережеві інженери оцінюють існуючу мережеву інфраструктуру, поверхові плани, продуктивність додатків і мережі, вимоги до потужності та багато інших факторів.

Цей етап оцінки допомагає мережевим професіоналам зрозуміти, де розмістити точки доступу (AP), визначити, які бездротові технології та обладнання використовувати, наприклад Wi-Fi, стільниковий зв'язок, бездротові сітчасті мережі, шлюзи та бездротові маршрутизатори, а також визначити потенційні проблеми в мережі.

Як розпочати планування та проектування бездротової мережі

Хоча підприємства мають різні вимоги до бездротового зв'язку для задоволення потреб бізнесу, найкращі методи планування бездротової мережі залишаються незмінними. Нижче наведено дев'ять основних напрямків, які команди мереж повинні враховувати, починаючи процес планування бездротового зв'язку.

1. Покриття

Перший, певною мірою очевидний етап у плануванні бездротової мережі полягає у визначенні того, де організація потребує покриття. Покриття стосується географічного розташування, яке отримує бездротовий доступ. Організації з

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 16 |

більшими або розподіленими місцями мають ширші вимоги до покриття, ніж організації з меншими або локалізованими сайтами.

На цьому кроці мережеві команди розглядають географічні аспекти, наприклад розташування сайтів, а також оцінюють екологічні та сезонні фактори, сказав Мартін. Наприклад, команди повинні оцінити мережу під час пікових ділових навантажень, як-от свята чи особливі події, щоб визначити ідеальні вимоги до продуктивності протягом року.

Команди також повинні звернути увагу на складні зони покриття, які можуть мати товсті стіни, металеве середовище, охолодження або інші фактори, які можуть створювати перешкоди для радіочастот (РЧ) і точок доступу.

2. Ємність

Ємність означає, скільки трафіку може підтримувати бездротова мережа, враховуючи швидкість, затримку та вузькі місця. Середовища з великою місткістю, такі як стадіони чи концертні майданчики, повинні підтримувати багато користувачів одночасно, часто з різними типами бездротових пристроїв.

Під час цього кроку мережеві команди оцінюють очікуване зростання своїх організацій, з'ясовують, скільки користувачів зараз у мережі та приблизну кількість, яка може приєднатися найближчим часом. Цей процес допомагає мережевим групам визначити типи клієнтів, які має підтримувати бездротова мережа, очікувану пропускну здатність, кількість точок доступу для розгортання та необхідні швидкості передачі даних.

Команди повинні без вагань запитувати про можливість будь-яких придбань бізнесу, які могли б додати нових користувачів мережі. За словами Мартіна, на цьому етапі команди також повинні почати вивчати типи додатків, які мережа повинна підтримувати зараз і в наступні кілька років.

3. Додатки

Третій етап полягає в глибшому зануренні у вимоги організації до програм. Мережеві команди повинні робити все можливе, щоб мережа відповідала бізнес-пріоритетам, включаючи критично важливі програми та робочі

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 17 |

процеси.

Фактори застосування включають наступне:

- Програма розміщена локально чи в хмарному середовищі?
- Це голосова чи відеопрограма, яка потребує низької затримки?
- Це програма, яка не пов'язана з роботою, наприклад Netflix або YouTube?

YouTube?

- Як офісні та віддалені співробітники отримують доступ до програми?

Ці міркування також включають мобільну підтримку, оновлення політики щодо робочої сили, маркетингові ініціативи та гостьовий доступ до Wi-Fi, сказав Мартін. Крім того, команди не повинні забувати про оцінку програм, які потребують великої кількості даних і пропускної здатності, як-от Інтернет речей і потокове відео, і відповідно планувати потужність і політики.

4. Безпека

Хоча вимоги та проблеми безпеки може бути важко передбачити, організації виграють, коли команди вживають заходів для захисту бездротової мережі, сказав Мартін. На цьому етапі команди вивчають такі функції безпеки, як запобігання та виявлення бездротових вторгнень, двофакторна автентифікація, рольовий доступ і брандмауери.

Цей етап планування включає наступні кроки:

1. Оцініть протоколи бездротової безпеки, такі як Wired Equivalent Privacy та різні версії Wi-Fi Protected Access.
2. Налаштуйте суворі політики безпеки для автентифікованих користувачів, гостей і програм.
3. Створіть надійний пароль і стратегію багатфакторної автентифікації.
4. Запустіть оновлення програмного забезпечення точки доступу.

Крім того, команди повинні бути в курсі різних типів бездротових атак, таких як військове водіння, злий близнюк і викрадення бездротової мережі. З міркувань відповідності Мартін сказав, що деякі бездротові системи автоматично реєструють дані для аудиту, щоб мінімізувати ручні записи або аналіз.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 18 |

5. Простота

П'ятий етап у плануванні бездротової мережі полягає в спрощенні, особливо при розгляді управління. Мережні команди отримують переваги від автоматизації, яка допомагає у вирішенні несправностей і скорочує час, витрачений на рутинні завдання керування.

Мартін порадив командам оцінити системи керування, які пропонують графічний інтерфейс користувача зі зручною навігацією та надають детальну інформацію про точки доступу, фальшиві пристрої та реєстрацію гостьової Wi-Fi, серед інших міркувань.

6. Надмірність

Далі IT-командам слід зосередитися на тому, як підтримати бізнес, якщо з мережею щось піде не так. На цьому етапі команди оцінюють параметри резервного копіювання, AP і шлях відновлення після відмови, високу доступність, конфігурацію контролера, живлення та інші фактори, на які може вплинути збій мережі або подія.

7. Мережева інтеграція

Цей етап планування бездротової мережі допомагає командам точно визначити наскрізну видимість мережі та потреби в інтеграції. Наприклад, точки доступу наступного покоління, які підтримують стандарт Wi-Fi 802.11ax, вимагають 60 кіловат потужності, але тільки нові стандарти Power over Ethernet (PoE) підтримують таку потужність.

Таким чином, командам слід звернути увагу на те, як PoE та масштабованість працюють із мережею, яку вони будують. Інші фактори мережевої інтеграції включають надійність мережі, продуктивність, висхідні канали доступу до мережі та пропускну здатність труби.

8. Менеджмент

Наступним кроком є подальше вивчення системи управління. Розглянемо, як платформа керування мережею обробляє всі різні мережеві компоненти, такі як точки доступу, комутатори та програмне керування WAN. Тип управління,

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 19 |

який обирає команда, залежить від розміру організації. Менші організації можуть працювати з кластерами AP, які пропонують менші накладні витрати на керування, тоді як великі організації отримують переваги від хмарного керування, яке забезпечує кращу масштабованість і резервування.

9. Огляд місця

Іншим важливим кроком у плануванні бездротового зв'язку є проведення опитування на місці. Бездротові обстеження об'єктів відрізняються за типом, з прогнозованими, пасивними та активними обстеженнями, доступними як апаратні або програмні засоби планування.

Команди можуть провести прогнозне обстеження місця перед створенням бездротового дизайну або якщо вони переїжджають на нові місця. Пасивне опитування сайту допомагає командам збирати інформацію про сайт після встановлення обладнання. Активне опитування сайту аналізує роботу бездротової мережі. Кожен тип опитування допомагає організаціям визначити положення точки доступу, радіочастоту, пропускну здатність, якість обслуговування, потенційні перешкоди та інші показники.

Висновок

Планування бездротової мережі є безцінним процесом, коли мережеві групи проектують бездротову мережу. Оцінюючи такі фактори, як ємність, покриття, вимоги до додатків, безпека та шифрування, простота, резервування, інтеграція, керування та огляди сайтів, команди можуть створити бездротовий дизайн, який найкраще підтримує їхній бізнес, співробітників і клієнтів.

3.2 Розробка структурної схеми

Бездротова мережа вашого бізнесу розширює можливості майже для кожного відділу, продукту, послуги та співробітника. Створення та підтримка успішної мережі Wi-Fi має бути однією з головних ІТ-цілей вашої організації.

Першим кроком до чудової бездротової мережі є чудовий дизайн. Проект

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 20 |

Wi-Fi служить схемою для вашої бездротової мережі. Витративши час на розуміння компонентів чудового дизайну та дотримуючись найкращих практик, ви не лише підвищите надійність і продуктивність своєї мережі, але й заощадите гроші, не витрачаючи надмірно на непотрібну мережеву інфраструктуру.

Наявність правильних інструментів для правильного проектування є основоположним для розгортання та обслуговування сучасних критично важливих бездротових мереж.

Що таке Wi-Fi Design?

Проектування Wi-Fi – це процес, який відповідає вимогам вашого бізнесу щодо бездротового підключення та перетворює їх на високопродуктивну та надійну мережу Wi-Fi. Це переклад ваших бізнес-потреб – скільки пристроїв потребує Wi-Fi (ємність) і де він їм потрібен (покриття) – у план розгортання, у якому детально вказано, скільки точок доступу вам знадобиться, де їх потрібно встановити та як їх слід налаштувати, щоб задовольнити вимоги ваших користувачів. Кінцевим результатом є надійний Wi-Fi для всіх пристроїв і програм у вашій мережі, а також щасливі співробітники та клієнти.

Чому дизайн Wi-Fi важливий

Бездротові мережі схожі на електрику в сучасному діловому світі. Коли Wi-Fi не працює, робота припиняється. Поганий дизайн може не тільки негативно вплинути на продуктивність сьогодні, але й мати довгострокові наслідки, ускладнюючи усунення несправностей і вирішення проблем пізніше. Дорогі збої, втрата прибутку та низька продуктивність можуть бути результатом поганого дизайну Wi-Fi.

Професійні інструменти проектування Wi-Fi дозволяють моделювати та перевіряти ефективність вашого проекту та гарантувати, що він досягне високої продуктивності з першого дня, а також дозволяють повторювати та оптимізувати для врахування змін ваших вимог з часом. Коли ви створюєте проект бездротової мережі – і ви отримуєте його правильно з першого разу – ви значно покращуєте загальну продуктивність Wi-Fi.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 21 |

Хороший Wi-Fi – це добре для бізнесу. Wi-Fi є частиною майже всіх критично важливих програм, які використовуються в бізнесі – від автономних навантажувачів на складах до парку ноутбуків і телефонів, розміщених у вашій організації. Це в медичних пристроях, що підтримують життя, на які ми покладаємося для здоров'я та благополуччя. Це тримає нас на зв'язку, коли нам потрібно спілкуватися та співпрацювати, і дає нам можливість втекти, коли нам потрібно ізолюватися.

Ви помічаєте Wi-Fi, лише коли він не працює. Усі ми відчували розчарування через погану роботу мереж Wi-Fi. Повідомляється, що в усьому світі 200 мільйонів бездротових мереж працюють погано – вони не можуть задовольнити вимоги щодо загального покриття та пропускної здатності з мінімальними перешкодами. Однією з основних причин поганої роботи мереж Wi-Fi є поганий дизайн Wi-Fi. Оскільки люди та організації все більше покладаються на мережі Wi-Fi, ризик їхньої несправності стає дедалі серйознішим.

У добре спроектованих бездротових мережах легше виявляти несправності та виправляти їх. Чудовий Wi-Fi починається з дизайну, який не тільки створює чудову базову лінію для вашої мережі, але й дозволяє легко виконувати регулярні перевірки стану та виявляти перешкоди, перш ніж вони стануть масовими проблемами.

6 ключових міркувань дизайну для задоволення вимог вашого бізнесу та РЧ

Найважливішим аспектом проектування високопродуктивної мережі Wi-Fi є визначення вимог до мережі. Дуже часто мережі не відповідають вимогам через неповний процес виявлення.

Подумайте про це так: якби хтось сказав вам піти будувати будинок, ви б почали з того, щоб завантажити 2x4s? Ні. Спочатку ви поставили б багато запитань – наскільки великий будинок ми будемо, чи існують якісь обмеження на кількість, який стиль будинку є кращим тощо. Будуючи мережу Wi-Fi, ви

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 22 |

також хочете спочатку поставити запитання та задокументувати повний список вимог. Це вбереже вас від помилок і головного болю.

Вимоги можна розділити на дві основні категорії: 1) бізнес-вимоги та 2) вимоги до РЧ.

Вимоги до бізнесу

Визначення бізнес-вимог до того, як буде використовуватися мережа, полегшує перетворення бізнес-потреб у конкретні вхідні дані для вашого програмного забезпечення для проектування.

Які різні типи клієнтів знадобляться для підключення до вашої мережі? Скільки з цих клієнтів потребують одночасного доступу? Який найменш потужний і найважливіший пристрій для вашого бізнесу? Які очікування від мережі, крім «Нам потрібен хороший Wi-Fi»? Відповіді на ці запитання допоможуть вам перетворити бізнес-потреби на вимоги до дизайну Wi-Fi щодо покриття, ємності та найменш потужного, найважливішого пристрою.

1. Покриття

Одним із найважливіших аспектів проектування Wi-Fi є планування покриття. Основне покриття залежить від зони та оптимізації відстані навколо ваших бездротових передавачів, щоб забезпечити достатню потужність сигналу для підключення пристроїв з підтримкою Wi-Fi. Багаторівневе ефективне вторинне покриття забезпечує потрібну кількість перекриття, щоб полегшити роумінг пристроїв і забезпечити резервування для критично важливих для бізнесу потреб Wi-Fi.

Поганий дизайн може призвести до занадто великої кількості точок доступу (що може збільшити загальні витрати на апаратне забезпечення та встановлення та може спричинити суперечку/перешкоди CoChannel) або занадто мало точок доступу (що не забезпечить необхідні вимоги до покриття та призведе до розривів покриття).

Інструменти проектування Wi-Fi, дають чітке уявлення про покриття та потужність сигналу, дозволяючи вам змінювати розташування та конфігурації

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 23 |

точки доступу на льоту та візуалізувати, як саме ці зміни можуть вплинути на покриття в середовищі.

2. Ємність

Планування ємності виходить за рамки охоплення та враховує різні типи та кількість пристроїв і програм, які підключатимуться до мережі. Пропускна здатність бездротової мережі – це вимірювання обсягу трафіку, що одночасно підтримується бездротовою мережею, на основі споживаної пропускної здатності.

Погано сплановані вимоги до потужності можуть бути руйнівними для користувачів. Низька швидкість і періодичні перерви з'єднання можуть бути результатом неправильного визначення вимог до використання. Це також може бути симптомом зростаючого болю, оскільки з часом додається більше користувачів і з'являються нові пристрої без коригування збільшеного попиту на ємність.

Потреби в ємності також можуть відрізнятися для різних областей сайту залежно від вашого випадку використання. Візьмемо, наприклад, готелі. Гостьові номери, лобі, відкритий басейн і конференц-центр можуть мати унікальні вимоги до місткості – і програмне забезпечення для проектування Wi-Fi, спрощує проектування зон різної місткості відповідно до унікальних потреб кожної зони.

3. Проектування для найменш потужного, найважливішого пристрою

Переглядаючи різні типи пристроїв, які під'єднуюватимуться до вашої мережі, важливо визначити, які пристрої є найбільш критичними, а які з цих пристроїв є найменш технологічно просунутими – ці пристрої відомі як найменш здатні, найважливіші пристрої (LCMID).

Вірте чи ні, але розробити Wi-Fi для найновіших пристроїв, що з'явилися на ринку, як правило, досить просто. Потрібно визначити єдиний пристрій, який, якби він раптово відключився, призупинив би бізнес – це складна частина.

Ось кілька звичайних підозрюваних щодо LCMID вашої мережі:

– 10-річний складський сканер витрачав 12 годин на день на сканування

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 24 |

штрих-кодів для управління запасами.

– Реєстри торгових точок, які використовуються для полегшення роздрібних операцій.

– Ноутбук вашого генерального директора (просто відмовляється брати новий).

Для цих типів пристроїв вам потрібно вивчити опубліковані виробником специфікації, щоб переконатися, що вони надійно працюють у мережі. Ваш прогнозний дизайн настільки хороший, наскільки вхідні дані визначені вами, і визначення вашого LCMID має вирішальне значення для дизайну вашої мережі Wi-Fi.

Вимоги РЧ

Фізичне середовище відіграє велику роль у роботі мережі. Поверніться до поверхового плану майданчика та пройдіться сайтом, щоб зібрати інформацію, яка допоможе вам визначити поведінку радіочастот (РЧ) у вашому оточенні.

Наскільки висока стеля в зоні покриття? Чи є достатній доступ для підключення точок доступу? З чого зроблені стіни? Наскільки шумні сусідні мережі? Відповіді на ці запитання допоможуть вам перевести чинники навколишнього середовища у вимоги до радіочастот для перешкод у фізичному середовищі, затухання матеріалу стін та активності радіочастотного спектру.

4. Перешкоди у фізичному середовищі та місця встановлення точок доступу

Високі стелі, відкриті металеві повітропроводи, коливання інвентарю, житлові атріуми та інсталяції сучасного мистецтва можуть не бути задокументовані на простому плані поверху будівлі, але такі перешкоди слід враховувати у ваших вимогах до Wi-Fi.

Плани поверхів розповідають лише частину історії. Якщо це можливо, ви повинні пройти сайт і зібрати інформацію, яка допоможе вам визначити поведінку радіочастот у вашому оточенні.

Виконання попереднього обстеження підлоги допоможе вам отримати

| | | | | | | | |
|------|------|----------|--------|------|--|----------------------------------|------|
| | | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | | 25 |

правильну інформацію для підключення до програмного забезпечення прогнозного проектування. Переконайтеся, що ви задокументували будь-які потенційні занепокоєння щодо радіочастот: відкриті стелі з повітроводами, колонами, вивісками, великими предметами меблів, забороненими зонами тощо. Ці покрокові вказівки також можуть висвітлити раніше невраховані обмеження щодо розміщення бездротової інфраструктури – де ви не можете розмістити точки доступу або де ви не можете прокласти кабелі.

5. Випробування матеріалу стін і затування

Розмір, форма та типи стінових матеріалів у вашому мережевому середовищі мають бути враховані під час проектування Wi-Fi. Це тому, що фізичні характеристики середовища впливають на радіочастотне покриття.

Кожна стіна послаблює сигнал Wi-Fi. Це означає, що РЧ-сигнал частково або повністю поглинається матеріалом. Гіпсокартон зазвичай знижує потужність сигналу на 3 дБ. Великі бетонні стовпи можуть зупинити сигнал Wi-Fi! Розуміння різних матеріалів у вашому середовищі та їх значення затування є ключовим для проектування чудової бездротової мережі.

Використання діагностичного та вимірювального пристрою світового класу забезпечить вам точні радіочастотні вимірювання, необхідні для вашого проекту. Перевірівши типи стін, ви або підтвердите, що ваш прогнозний проект правильний, або матимете можливість відкоригувати на основі емпіричних даних, які ви зібрали.

6. Активність радіочастотного спектру

Ваша мережа Wi-Fi живе у світі електромагнітного спектру. Розуміння активності спектру навколо вас веде вас до ефективного плану каналів для вашого проекту. Ось кілька речей, які слід враховувати:

- Конкурентність каналів: точки доступу, як у мережі вашого сусіда, так і у вашій власній, потрібно розташувати належним чином із відповідними планами каналів, інакше існує ризик конфлікту каналів.
- Перешкоди, не пов'язані з Wi-Fi. Такі речі, як мікрохвильові печі,

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 26 |

пристрої Bluetooth, шпигунські камери та датчики руху, можуть перешкоджати здатності вашої мережі Wi-Fi надсилати й отримувати дані. Проблеми, спричинені перешкодами Wi-Fi, можуть варіюватися від періодичної втрати з'єднання до зниження швидкості передачі даних і мережі до зниження потужності сигналу.

– Перевірки DFS і радар: залежно від використовуваних частот радарне обладнання може заважати передачі даних мережі Wi-Fi і навпаки. Радар може вплинути на продуктивність вашої мережі Wi-Fi, обмежуючи кількість каналів 5 ГГц, які ви можете використовувати, або він може спричинити зниження продуктивності, якщо ваші точки доступу постійно змінюють канали, щоб уникнути подій DFS.

– Ширина каналу: чим ширший канал, тим вища потенційна пропускна здатність. Залежно від поточного радіочастотного середовища та щільності радіостанцій Wi-Fi ви зможете визначити бажану ширину каналу для свого проекту. Завжди використовуйте максимально можливу ширину каналу, не спричиняючи надмірних проблем із конкуренцією.

Трансляція вимог у проект бездротової мережі

Чудова мережа Wi-Fi починається з чудового дизайну, а чудовий дизайн починається з точних вимог Wi-Fi. Ставлячи правильні запитання та визначаючи шість вимог, представлених у цьому посібнику, ви зможете розробити прогнозний дизайн, який ідеально відповідає потребам вашого бізнесу. Ось короткий аналіз того, як бізнес-вимоги перетворюються на входи програмного забезпечення для розробки:

– Покриття: розміщення точок доступу та візуалізація покриття на масштабованому плані поверху з точними стінами

– Ємність: ідентифіковані профілі використання та пристроїв зі списком програм і моделей клієнтів, що використовуються

– Найменш здатний, найважливіший пристрій: профіль пристрою, створений для LCMID

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 27 |

- Перешкоди у фізичному середовищі: встановлена висота стелі та примітки щодо розгортання, наведені для врахування перешкод
- Послаблення матеріалу стіни: відповідні типи стін, що використовуються на всьому плані поверху, включаючи створені на замовлення типи стін
- Діяльність радіочастотного спектру : план каналу, який зменшує перешкоди в одному каналі та оптимізує продуктивність клієнта

Висновок

Тепер, коли ви дізналися про найкращі методи проектування бездротової мережі, ви на шляху до прийняття кращих рішень під час створення мережі Wi-Fi.

Ключові елементи надійної та надійної стратегії проектування мережі Wi-Fi мають включати такі міркування.

Що потрібно знати у 2025 році, якщо ви відповідатимете за проектування мережі Wi-Fi?

Плануйте ємність, а не лише покриття

Не так давно проектування мережі Wi-Fi зосереджувалося на фізичних оглядах сайтів, щоб визначити кількість точок доступу, необхідних для забезпечення достатнього покриття. Після цього ви оціните результати та порівняєте кількість точок доступу з прийнятною мінімальною потужністю сигналу, і вся конструкція WLAN буде визнана успішною. Цей шлях підходить для мереж WLAN, які планується охопити, але, звичайно, не є правильним підходом для задоволення вимог щодо пропускної здатності.

У цьому методі такі ключові елементи, як кількість одночасних користувачів і потреби програми в пропускній здатності, не враховуються. Інженери з бездротового зв'язку та IT-консультанти повинні повністю розуміти вимоги до проектування мережі WiFi, щоб забезпечити успішне проектування. Це допоможе зменшити потребу в подальших оглядах сайту після розгортання інфраструктури WiFi у довгостроковій перспективі.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 28 |

Основні моменти, які слід враховувати:

- Типи програм, які очікуються в мережі, наприклад, веб-перегляд, VoIP-дзвінки, програмне забезпечення або потокове відео.
- Технології, які підтримує інфраструктура WiFi (802.11 a/b/g/n/ac).
- Кількість клієнтських пристроїв, які підключаються до мережі WiFi одночасно (допомагає визначити кількість просторових потоків, технологію та типи точок доступу).
- Ключові географічні зони, які потрібно охопити та надати Wi-Fi, а також кількість одночасних пристроїв на територію.
- Обмеження потужності. Набагато корисніше мати інфраструктуру, оснащену PoE+, яка дозволяє підтримувати високопродуктивні точки доступу.

Використання каналу

Платформа керування мережею Wi-Fi повинна була включати інструмент для керування радіочастотами. Таким чином, він може динамічно призначати канали точок доступу, регулювати потужність передачі точки доступу та забезпечувати пом'якшення розриву покриття для інфраструктури WiFi.

Наприклад, для стандарту бездротового зв'язку 802.11ac керування радіочастотами має виконуватись із шириною каналу 20, 40 і 80 МГц. Різні клієнтські пристрої підтримуватимуть різну ширину каналу для протоколів 802.11. Клієнтські пристрої, які підтримують більшу ширину каналу, підтримуватимуть вищу пропускну здатність у межах певного протоколу.

Оцініть, скільки клієнтських пристроїв можна виділити на діапазон

Завдяки новітнім технологіям більше клієнтських пристроїв тепер підтримують дводіапазонну роботу, і, отже, за допомогою власних пристроїв реалізації можна керувати на 5 ГГц. Типовий підхід до проектування полягає в розподілі 30/70 між 2,4 ГГц і 5 ГГц.

Щоб дізнатися більше про те, як WiFi з програмним керуванням перетворює бізнес, прочитайте нашу статтю.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 29 |

Думайте про мобільні пристрої – знову думайте про мобільні пристрої

Також для мобільних пристроїв необхідно створити гарну мережу WiFi – це дуже важливо для успіху розгортання будь-якої мережі WiFi останнім часом. Бездротовий дизайн для розгортання має бути оптимізований для кожного пристрою, від смартфонів і Інтернету речей до комп'ютерів і планшетів. Правильний дизайн бездротового зв'язку стоїть на першому місці, особливо для забезпечення високої продуктивності пристрою та загальної мобільності для кращого досвіду кінцевого користувача. Це також означає врахування таких функцій, як 802.11r/w/v.

SSID

Щоб максимізувати продуктивність у бездротовому просторі та спростити розгортання, спробуйте звести до мінімуму кількість SSID, що транслюються в середовище. Недоліком увімкнення більше SSID є те, що це створює додаткове використання каналу через накладні витрати. Цільове значення трьох SSID на точку доступу забезпечує гнучку, але просту модель розгортання.

Наприклад, ви можете мати один відкритий SSID для всіх некерованих пристроїв (щоденні гості, консультанти, BYOD, клієнти..) з автентифікацією на порталі авторизації. Другий SSID для автентифікованих користувачів і пристроїв 802.1x. І третій SSID для конкретних випадків використання або спеціалізованих бездротових пристроїв, наприклад, VoIP-телефонів з підтримкою Wi-Fi, пристроїв, що не підтримують 802.1x, або спеціалізованих мережевих пристроїв.

BYOD

Користувачі хочуть підключати свої особисті пристрої до публічних і приватних мереж WiFi. Це стандарт. Просто переконайтеся, що користувачі направляються через фільтр веб-вмісту, щоб забезпечити безпечний досвід перегляду для всіх користувачів. Крім того, оскільки задіяно кілька пристроїв, рішення BYOD має відповідати місцевим і глобальним правилам конфіденційності даних. Щоб зрозуміти, як конфіденційність даних змінила цифровий бізнес за останні кілька років, перегляньте наш останній звіт.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 30 |

Нарешті, обмеження споживання смуги пропускання для кожного користувача є важливим у мережі для керування її продуктивністю. Важливо взяти до уваги, що тенденція BYOD безпосередньо впливає на пропускну здатність і пропускну здатність.

ІОТ

Поява Інтернету речей (IoT) ускладнила проектування мережі WiFi. У зв'язку зі збільшенням кількості взаємопов'язаних пристроїв, починаючи від датчиків і інтелектуальних пристроїв і закінчуючи промисловим обладнанням, надійна мережа WiFi повинна бути здатна обробляти збільшений трафік і унікальні вимоги до пристроїв IoT. Стратегія проектування повинна включати такі міркування, як щільність пристроїв, пропускну здатність даних і затримка, щоб забезпечити безперебійне підключення та оптимальну продуктивність для програм IoT. Крім того, необхідно вжити заходів безпеки, щоб захистити конфіденційні дані, що передаються між пристроями IoT і мережею. Інтегруючи елементи IoT у проект мережі WiFi, організації можуть використовувати весь потенціал цієї трансформаційної технології та відкрити нові можливості для автоматизації, моніторингу та ефективності.

Обмеження пропускну здатності

Останньою рекомендацією щодо кращого дизайну мережі WiFi є встановлення обмеження пропускну здатності кожного клієнтського пристрою для всього трафіку мережі WiFi. Пріоритет має бути наданий таким програмам, як відео та голос. Наприклад, 5 Мбіт/с є хорошою рекомендацією для обмеження пропускну здатності для кожного клієнта в середовищі з високою щільністю.

З огляду на те, що численні режими підключення та типи пристроїв стають звичайною нормою в офісах, готелях, гостьових будинках тощо, не буде помилкою сказати, що погано спланована та реалізована мережа Wi-Fi може негативно вплинути на IT-продуктивність усієї організації. У деяких ситуаціях це може бути навіть шкідливим і перешкоджати критичним бізнес-процесам. Тому команди IT-інфраструктури повинні мати на увазі всі ці параметри, щоб думати

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 31 |

на довгострокову перспективу та розробити надійну, сумісну та безпроблемну мережу WiFi.

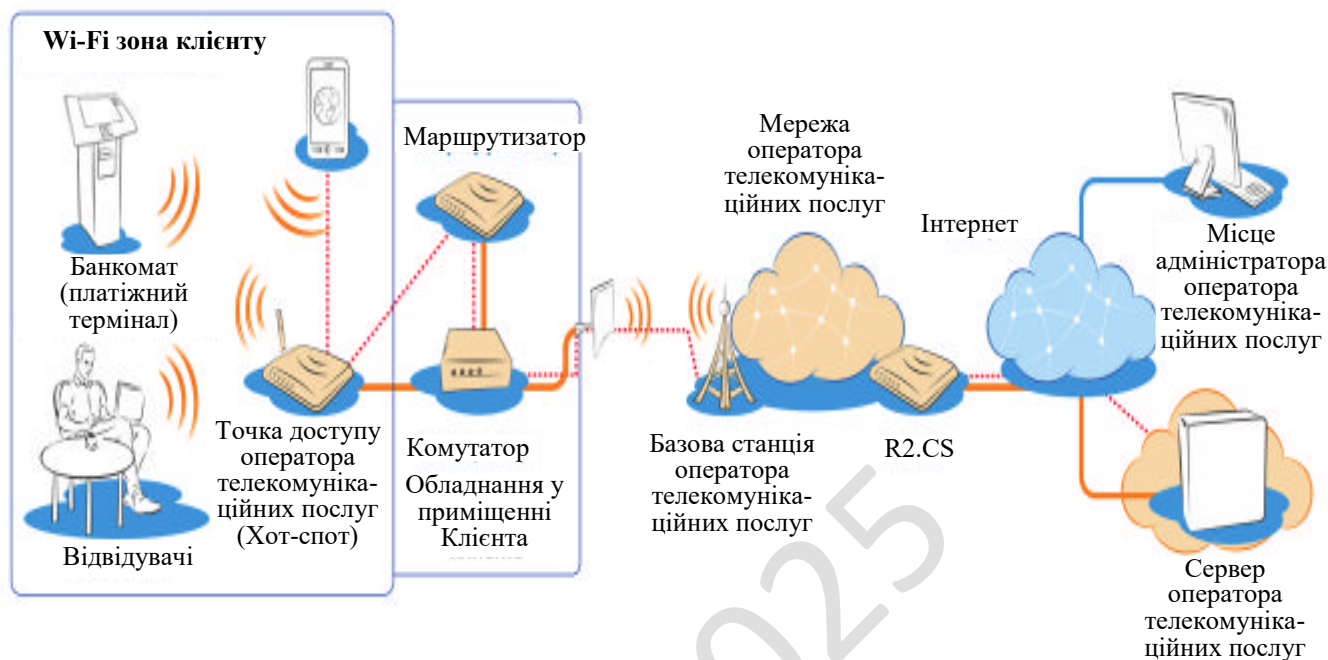


Рисунок 3.1 – Структурна схема системи

3.3 Розробка функціональної схеми

Розвиток технологій бездротового зв'язку стало спонукальним фактором для повторного аналізу, а в деяких випадках і для перегляду існуючих стандартів, що визначають топологію й інші аспекти кабельних систем, використовуваних для підключення точок доступу. Розглянемо деякі із цих стандартів.

Багато стандартів, що стосуються бездротових мереж, концентруються на питаннях підключення точок доступу до загальної мережі оператора телекомунікаційних послуг об'єкта. Суточніше всього вибрати розмір стільники таким чином, щоб побудована мережа WLAN забезпечувала ефективну підтримку додатків IEEE 802.11n і 802.11ac, а кабельна розподільна мережа гарантувала достатню продуктивність сьогодні й у майбутньому.

В 2004 році міжнародні організації ISO і IEC представили технічний звіт TR-24704, що сфальцьований на питаннях «універсальної кабельної інфраструктури для підключення бездротових точок доступу усередині приміщення». Складений у період становлення корпоративних систем Wi-Fi, він містить чудове передбачення росту можливостей сучасних бездротових технологій.

Установлюючи стандарти кабельної інфраструктури для бездротового доступу, звіт TR-24704 пропонує оптимальну схему розміщення точок бездротового доступу. Як модель мережі оператора телекомунікаційних послуг пропонується масив із прилягаючий друг до друга шестикутних стільників (осередків). Крім цього, передбачається використання всеспрямованих антен, із круговим випромінюванням у кожної соте. При правильній розстановці досягається мінімальне й одночасно оптимальне перекриття сусідніх стільників.

Для досягнення більше високої ємності зона покриття кожного стільника обмежений радіусом 12 м. TR-24704 рекомендує розміщати телекомунікаційні розетки якнайближче до центра стільники. Це забезпечує максимальну гнучкість при установці окремих точок доступу й найкраще покриття.

Незабаром після появи документа TR-24704, Асоціація TIA представила свої рекомендації з реалізації кабельної інфраструктури для підключення бездротових точок доступу. Документ TIA TSB-162 припускає створення сітки із квадратних осередків зі стороною 18 м. Такий підхід більше відповідає типовому плануванню в Північній Америці, він спрощує проектування й інсталяцію. Точки доступу розміщаються при можливій довжині шнура підключення до 13 м. Передбачаючи прийняття стандарту IEEE 802.11ac, укладачі нової редакції цього документа рекомендували використовувати кабельну проводку Категорії 6A, щоб забезпечити достатню пропускну здатність і підтримку систем Power-over-Ethernet (PoE) зі збільшеною потужністю.

До кінця 2012 року Асоціація TIA прийняла ще один стандарт розраховуючи на високу щільність розміщення точок доступу. Документ TIA-

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 33 |

4966 призначений для забезпечення бездротового покриття в освітніх установах, але викладені в ньому рекомендації корисні для будь-яких великих приміщень або будинків з високою концентрацією бездротових клієнтів.

При оснащенні великих відкритих приміщень рекомендується при виборі щільності розміщення точок доступу враховувати передбачуване число користувачів. Плануючи покриття для приміщень із декількома секціонованими зонами, необхідно виходити з їхньої площі. У типовому офісному будинку рекомендується одна точка доступу на кожні 230 м². На менш «дружніх» для радіосигналу об'єктах, наприклад у гуртожитках, пропонується встановлювати по одній точці доступу на кожні 150 м².

Планування бездротової мережі оператора телекомунікаційних послуг доступу

Незважаючи на різницю в рекомендаціях, які пропонуються в зазначені вище стандартах, кожний з них може допомогти ІТ-фахівцям у проектуванні ефективної WLAN з високою ємністю. Але, звичайно, для цього потрібно й більше глибоке розуміння специфіки даного процесу.

Щоб повною мірою використовувати переваги таких технологій, як 802.11ac, необхідно враховувати електромагнітну обстановку на об'єкті, рівень перешкод і їхніх джерел, майбутню потребу в ємності мережі оператора телекомунікаційних послуг, вимоги до кабельної проводки й забезпечення електроживлення. У рамках цієї статті ми не зможемо дати вичерпний аналіз розглянутої тут теми, але постараємося представити основні рекомендації з побудови кабельної інфраструктури й викласти загальні ідеї планування й реалізації ефективної мережі оператора телекомунікаційних послуг точок доступу.

В ідеалі розробка кабельної мережі оператора телекомунікаційних послуг і аналіз радіопокриття повинні здійснюватися в комплексі – це дозволить забезпечити максимальну ємність і гнучкість для задоволення поточних і майбутніх потреб кінцевих користувачів. На практиці структури, описані в

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 34 |

документах TIA TSB-162-а й ISO/IEC TR 24704, мають явні переваги, особливо для застосування в нових будинках.

У новому будинку кабельна сітка (pre-cabled grid) для підключення точок доступу й структурована кабельна система для IT- і іншого встаткування можуть розроблятися й інсталюватися в те саме час. Після того як кабельна сітка змонтована, бездротова інфраструктура на об'єкті може бути встановлена в будь-який час, причому незручності для співробітників будуть мінімальними. При визначенні місця для установки точок доступу рекомендується проводити радіочастотне обстеження, що дозволить оптимізувати розташування ТД у кожної конкретної соти (осередку).

Радіочастотне планування

Для нових інсталяцій більшість виробників устаткування WLAN рекомендують проводити радіочастотне обстеження об'єкта. В умовах, коли запити до ємності мінімальні й немає яких-небудь спеціальних вимог, досить лише оцінити умови поширення радіохвиль. В інших випадках може знадобитися більше ретельний аналіз, у тому числі питань, пов'язаних з перешкодами, продуктивністю й безпекою. Оскільки новий стандарт 802.11ac припускає істотне збільшення числа одиниць мережного встаткування й обсягу трафіку, виконання ретельного аналізу об'єкта стає дуже важливим – необхідно переконатися в тому, що мережа буде повністю відповідати як поточним, так і майбутнім вимогам до бездротового зв'язку.

Використовуючи подібний інструментарій, наприклад Wi-Fi-аналізатор AirMagnet компанії Fluke Networks, можна змоделювати характеристики бездротової мережі оператора телекомунікаційних послуг на конкретному об'єкті, оптимізувавши розміщення точок доступу. Ця програма дозволяє ввести план об'єкта, указати місця установки точок доступу й побачити картину радіопокриття в порівнянні з реальною. Крім того, вона забезпечує генерацію звітів і може бути інтегрована із засобами керування WLAN, що робить її

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 35 |

використання ідеальним інструментом для проектування бездротової мережі оператора телекомунікаційних послуг.

Ослаблення сигналу може бути пов'язане з рядом факторів, включаючи наявність поглинаючих матеріалів, таких як шафи й устаткування, або структурних перешкод – бетонних стін або сталевих перебирань. Це пояснює різке падіння потужності сигналу на стороні об'єкта, протилежної тої, де встановлена точка доступу.

Гарне покриття забезпечене практично скрізь, де раніше (при наявності тільки однієї точки доступу) сигнал була слабкий. Таке розташування типово для рішень 802.11n, коли точки доступу використовують канали, що не перекриваються. Забезпечуючи непогане радіопокриття, така конфігурація має обмежені можливості для більше інтенсивного використання Wi-Fi.

Перший крок у плануванні високопродуктивної мережі оператора телекомунікаційних послуг – поділ об'єкта на сітку осередків (стілників) відповідно до рекомендацій TIA TSB-162-а або ISO / IEC TR 24704. Цю сітку можна пристосувати й для задоволення потреби в кабельній проводці з боку інших систем будинку. Такий інтегрований «відкритий» підхід (з певною волею вибору місця розміщення телекомунікаційних розеток) до реалізації СКС має як короткострокові, так і довгострокові переваги. При використанні сітки із квадратними осередками (згідно TSB-162-A) положення й щільність точок доступу можна міняти відповідно до заповнюваності конкретної області. Наприклад, у зоні з переговорними кімнатами буде потрібно кілька точок доступу, а в лабораторії можна обійтися їхнім мінімальним числом.

Зверніть увагу на те, що фактичне розміщення точок доступу ґрунтується на поточному використанні простору, тоді як грамотно спроектована кабельна проводка повинна забезпечити можливість майбутніх змін. По завершенні формування сітки осередків варто вибрати кабельну проводку, здатну підтримати майбутні потреби.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 36 |

Попереднє планування всієї проводки може значно спростити інсталяцію, а також наступну модернізацію, технічне обслуговування й ремонт мережі оператора телекомунікаційних послуг.

Розміщення й підключення точок доступу

Крім моделювання радіочастотного середовища й планування ємності, при проектуванні розміщення й підключення точок доступу необхідно продумати фізичну доступність, організацію розподільної мережі оператора телекомунікаційних послуг і електроживлення, а також загальну естетику інсталяції.

Фізична доступність. Грамотний вибір місця установки допоможе заощадити на обслуговуванні й модернізації й у цілому знизити експлуатаційні витрати. Дуже часто доступ до встаткування бездротового зв'язку ускладнений. Нерідко ТД розміщують над фальшпотолком або в закритих шафах, що вимагає прокладки кабелів у подпотолочному просторі. Але монтаж на стіну або на стелю все-таки переважніше. Це дозволяє завжди бачити індикатори стану ТД.

Треба ретельно підійти й до вибору місця розміщення телекомунікаційної розетки. Гарна доступність розетки, її розташування поруч із точкою доступу дозволять легко провести тестування лінії зв'язку й швидко відключити точку доступу для обслуговування або ремонту.

Розподільна мережа. У мережі оператора телекомунікаційних послуг традиційної топології телекомунікаційна розетка, використовувана для підключення ТД, прямо зв'язана горизонтальним кабелем з комутаційною панеллю, що перебуває в телекомунікаційній кімнаті на тому же поверсі. Альтернативний варіант – зонна архітектура, що може забезпечити легку установку, високу гнучкість і потенційно більше низькі експлуатаційні витрати.

У зонній моделі кабелі прокладаються від апаратної кімнати до конкретних виділених зон у будинку. Фіксована проводка закінчується на рівні точки консолідації (Consolidation Point, CP), звідки кабелями здійснюється підключення телекомунікаційних розеток для ТД. Такий підхід забезпечує

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 37 |

максимальну гнучкість при розміщенні першої розетки в кожному осередку, залишаючи необхідні ресурси для підключення додаткових розеток. Це може бути надзвичайно корисно в ході модернізації існуючих мереж: при грамотно обраному місці установки точки консолідації довгі джгути кабелів, що йдуть із телекомунікаційної кімнати, можуть бути зафіксовані в кабельних важкодоступних каналах. При наявності фіксованої проводки інсталювальники одержать «волю маневру» при прокладці подовжувальних кабелів для підключення до точки консолідації телекомунікаційних розеток, які можуть використовуватися для обслуговування ТД або іншого встаткування інтелектуального будинку.

Електроживлення. Якщо точки доступу будуть підключатися до звичайної електромережі оператора телекомунікаційних послуг, необхідно погодити всі дії зі службою головного енергетика, що у випадку високоплотних інсталяцій з використанням пленумного простору зробити досить складно. Із цієї причини більшість точок доступу сконструйовано для одержання електрики по слабкострумової СКС – за технологією PoE. Для забезпечення надійної роботи встаткування (особливо тих точок доступу, які встановлюються в закритих місцях або будуть експлуатуватися в складних кліматичних умовах) рекомендується перевірити канал PoE.

Вимоги до живлення нових точок доступу стандарту 802.11ac можуть виявитися більше твердими. Проектувальникам варто розглянути можливість прокладки щонайменше двох кабелів Категорії 6А до кожної точки доступу. Крім резервування каналу зв'язку, це допоможе організувати резервне живлення, і точка доступу зможе функціонувати навіть у випадку виходу з ладу одного з комутаторів або інжекторів PoE, до яких вона підключена.

Естетичні міркування. Бажано зробити все так, щоб кабельна проводка для точок доступу, включаючи телекомунікаційні розетки й шнури підключення, була схована або хоча б не впадала в око. Для цього намагаються задіяти елементи структури будинку, але коли неприпустимо торкатися унікальної

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 38 |

архітектури й/або дизайн інтер'єра, доводиться шукати вихід з положення. У кожному конкретному випадку деталі розміщення елементів інфраструктури повинні обговорюватися із клієнтами.

Функціональна схема розробленої системи зображена на рисунку 3.2. На ній показано, що структура системи складається з наступних блоків:

– Блок збирання статистичного навантаження на вузли мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi.

– Блок аналізу статистичного навантаження.

– Блок балансування навантаження у мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi.

– Блок формування звітів.

– Блок формування сигналів керування навантаженням.

Для рішення завдання вибору числа й розміщення елементів управляючих центрів (УЦ), представленого у вигляді інформаційно-обчислювального комплексу, запропоновано використовувати евристичні алгоритми, засновані на методах локальної оптимізації й теорії масового обслуговування. Альтернативний підхід до рішення завдання дискретного математичного програмування, заснований на теорії графів, неприйнятний через високу обчислювальну складність при великій кількості вузлів у мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi. Наприклад, у мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi з 10 вузлів існує 2^{45} варіантів розташування ліній зв'язку, включаючи безліч тривіальних випадків. Якщо припустити, що аналіз кожного варіанта становить 1 секунду, то на дослідження буде потрібно більш ніж $9 \cdot 10^8$ років.

На змістовному рівні завдання побудови оптимальної мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi з УЦ формулюється в такий спосіб. Виходячи із заданих значень інтенсивності запитів абонентів з урахуванням вводимих допущень і обмежень визначити оптимальні за критерієм мінімуму наведених витрат функціональні параметри мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi: число й розміщення програмно-апаратних модулів УЦ у пунктах мережі оператора телекомунікаційних

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 39 |

послуг побудованої на основі технології Wi-Fi, співвіднести групи абонентів з обслуговуючими їх УК УЦ, ємність УК УЦ і каналів зв'язку.

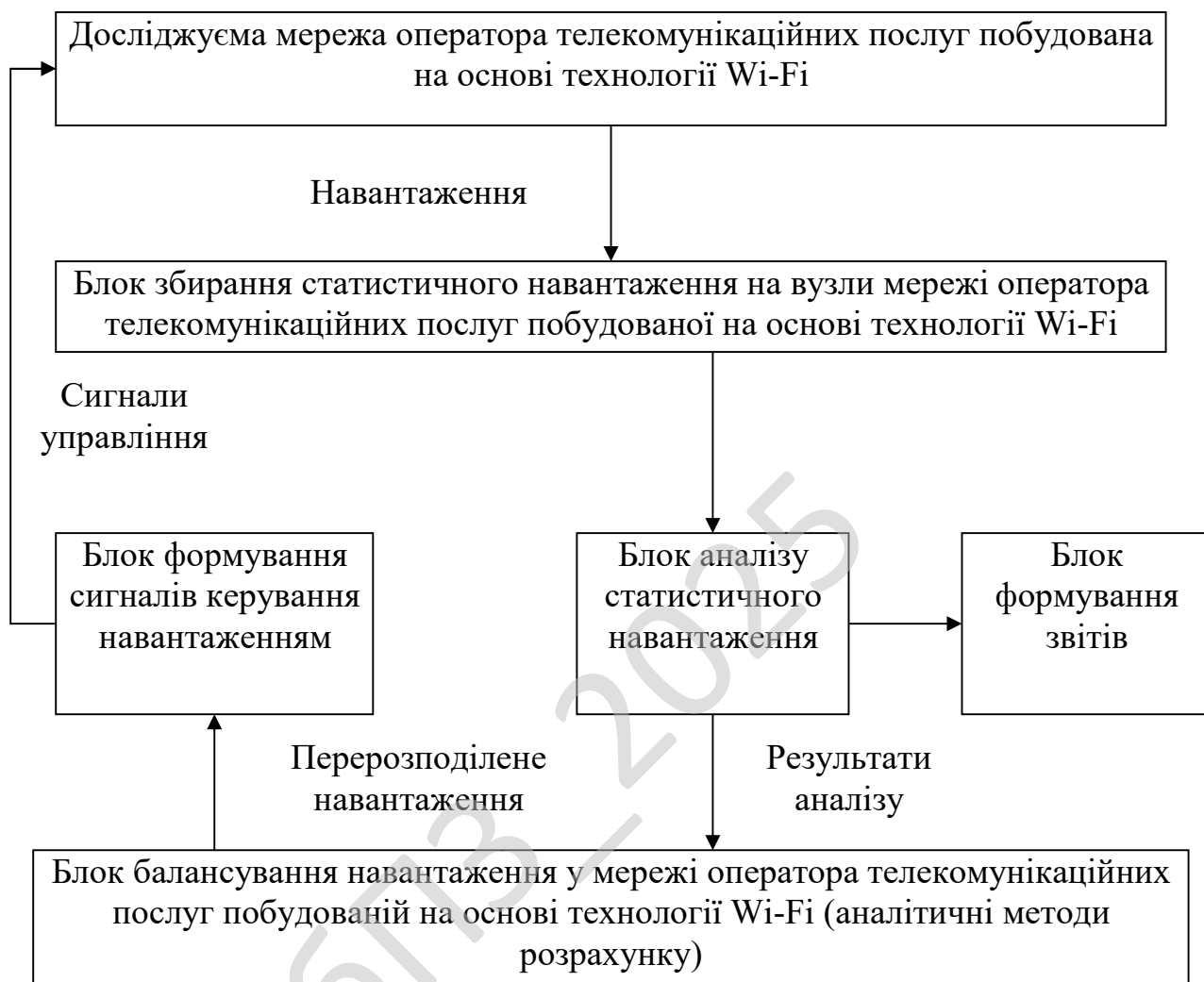


Рисунок 3.2 – Функціональна схема системи

При цьому повинні дотримуватися обмеження на якість обслуговування: середній час обробки повідомлення й імовірність своєчасного обслуговування не повинні перевищувати граничних значень.

Як критерій оптимізації обрані наведені витрати на канал зв'язку й вузол мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi.

Як модель, що інтерпретує інформаційний потік у мережі оператора телекомунікаційних послуг побудованій на основі технології Wi-Fi з УЦ, розглядається дві мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi масового обслуговування (СеМО):

- багатофазна СеМО з відмовами й повторними викликами;
- багатофазна СеМО комбінованою комутацією.

Завдання пошуку оптимальної структури мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi успішно вирішуються з використанням евристичних алгоритмів, заснованих на методах локальної оптимізації зі спрямованим перебором варіантів структури мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi.

На першому етапі вся безліч абонентських пунктів розбивається на групи, розташовувані в окремих зонах обслуговування УК УЦ. У процесі роботи алгоритму число УК УЦ (відповідно й чисто зон) змінюється від мінімального (наприклад, рівного одиниці), до максимального. Для кожного значення шукається локальне оптимальне розміщення УК УЦ. Абонентські місця приєднуються до найближчих УК УЦ.

Далі по черзі вибирається найкраще місце розташування кожного УК УЦ у межах своєї зони. Найкращий пункт для розміщення УК УЦ у межах своєї зони вибирається не із всіх можливих варіантів, а з пунктів підмножини A^* , число елементів якого істотно менше в порівнянні з безліччю A . Додатково прискорити процес перебору дозволяє уведення околиць, у які входять до десяти варіантів можливого розташування УК УЦ. Після УК УЦ δ -ї зони процес повторюється із УК УЦ першої зони. Якщо, починаючи з будь-якого УК УЦ, δ раз не вдалося поліпшити значення оптимізуемого функціонала, то процес вибору розміщення δ УК УЦ у фіксованих зонах припиняється, тому що досягається локальний екстремум.

При фіксованій кількості УК УЦ рішення завдання закінчується, коли перестановка УК УЦ не приводить до мінімізації оптимізуемого функціонала.

Після того як мережа оператора телекомунікаційних послуг побудована на основі технології Wi-Fi із УК УЦ сформована (знайдене місце розташування всіх УК УЦ і зроблене прикріплення абонентів), обчислюється значення параметрів оптимізації (середній час обслуговування T і ймовірність своєчасного обслуговування Q). Знайдені значення рівняються із граничними величинами (T_b , Q_L). Якщо обмеження не виконуються, то поточна структура виключається з розгляду.

Оптимальна структура мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi відповідає мінімальному значенню функціонала серед отриманих величин при різному числі УК УЦ.

У цьому розділі основним питанням є поліпшення якості функціонування мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi УЦ шляхом адаптивного балансування навантаження між його керуючими компонентами.

Серед аналізованих підходів по керуванню навантаженням і динамічним керуванням потоками інформації (метод рельєфів, матричний і хвильовий алгоритми, ігровий і ймовірнісно-ігровий методи) для імітаційного моделювання системи балансування навантаження на УК УЦ обраний матричний метод.

Аналіз вироблявся по наступних ознаках:

- вид аналізованої інформації;
- тип реакції на вхідний потік;
- тип системи керування;
- критерій оптимізації;
- обсяг службової інформації, що пересилається;
- часові витрати;
- складність реалізації.

Недоліком матричного методу для розглянутої прикладної області є критерій оптимізації у вигляді довжини шляхи комутації. Це можна усунути шляхом формування іншого критерію оптимізації, адаптованого під роботу

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 42 |

мережі оператора телекомунікаційних послуг побудованій на основі технології Wi-Fi з УЦ.

Матричний метод балансування навантаження орієнтований на алгоритми знаходження найкоротших шляхів. Даний метод дозволяє знайти довжини найкоротших шляхів між всіма вузлами мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi одночасно, ґрунтуючись на застосуванні операцій над матрицями відстаней. У розробленій модифікації методу пропонується замінити параметр довжини зв'язку інтегральним критерієм, розрахованим раніше.

У цьому випадку структуру мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi можна представити у вигляді матриці адаптивного балансування навантаження мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi, де показник оптимальності визначається як результат розподілу інтегрального критерію z -ого вузла (n_i) на показник доступності лінії зв'язку з h -ним вузлом.

Запропонована в роботі модифікація матричного підходу полягає в наступному:

- пропонується замінити параметр довжини зв'язку інтегральним критерієм;
- на кожному кроці побудови матриці балансування в додатковій матриці запам'ятовуються проміжні транзитні вузли. Це дозволяє по результатам багатокрокового перерозподілу навантаження видавати інструкції для УК УЦ по маршрутизації вступні на них повідомлень.

Запропоноване рішення по поліпшенню якості функціонування мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi з УЦ рівнялося з існуючими моделями якості обслуговування (Quality of Service). За результатами порівняння зроблений вивід про неможливість рішення завдання перерозподілу навантаження між УК УЦ винятково засобами цих моделей.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 43 |

Результатом моделювання є вказівки для УК УЦ, сформовані за допомогою програмного модуля розробленої системи, засобами якого вузли можуть перенаправляти надлишкове навантаження на аналогічні компоненти. З наведених результатів моделювання можна зробити вивід, що розроблена модель динамічного балансування навантаження мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi враховує основні характеристики роботи УЦ, дозволяючи оптимально перерозподілити вхідний інформаційний потік на її керуючих компонентах. У результаті балансування не тільки знімається надлишкове завантаження із УК УЦ, але й відбувається перерозподіл вхідного потоку з недоступних вузлів.

Матриця доступності, використовувана при моделюванні, дозволяє врахувати залежність від якості лінії зв'язку, що актуально, як при роботі через Інтернет-провайдерів у мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi загального користування, так і у випадку мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi корпоративного УЦ, що діє у ЛОМ або розподіленої VPN-мережі оператора телекомунікаційних послуг побудованої на основі технології Wi-Fi.

Модифікація матричного підходу знаходження оптимальних для прийняття надлишкового навантаження вузлів і введення інтегрального показника, дозволяють сформулювати нотації для перемикавання навантаження не тільки на вигідний УК УЦ, але й організувати маршрут перерозподілу, що складає з декількох транзитних вузлів.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 44 |

системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі. Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

- Процеси які являють собою трансформацію даних в рамках описуваної системи.
- Сховища даних (репозиторії).
- Зовнішні по відношенню до системи сутності.
- Потоки даних між елементами трьох попередніх типів.

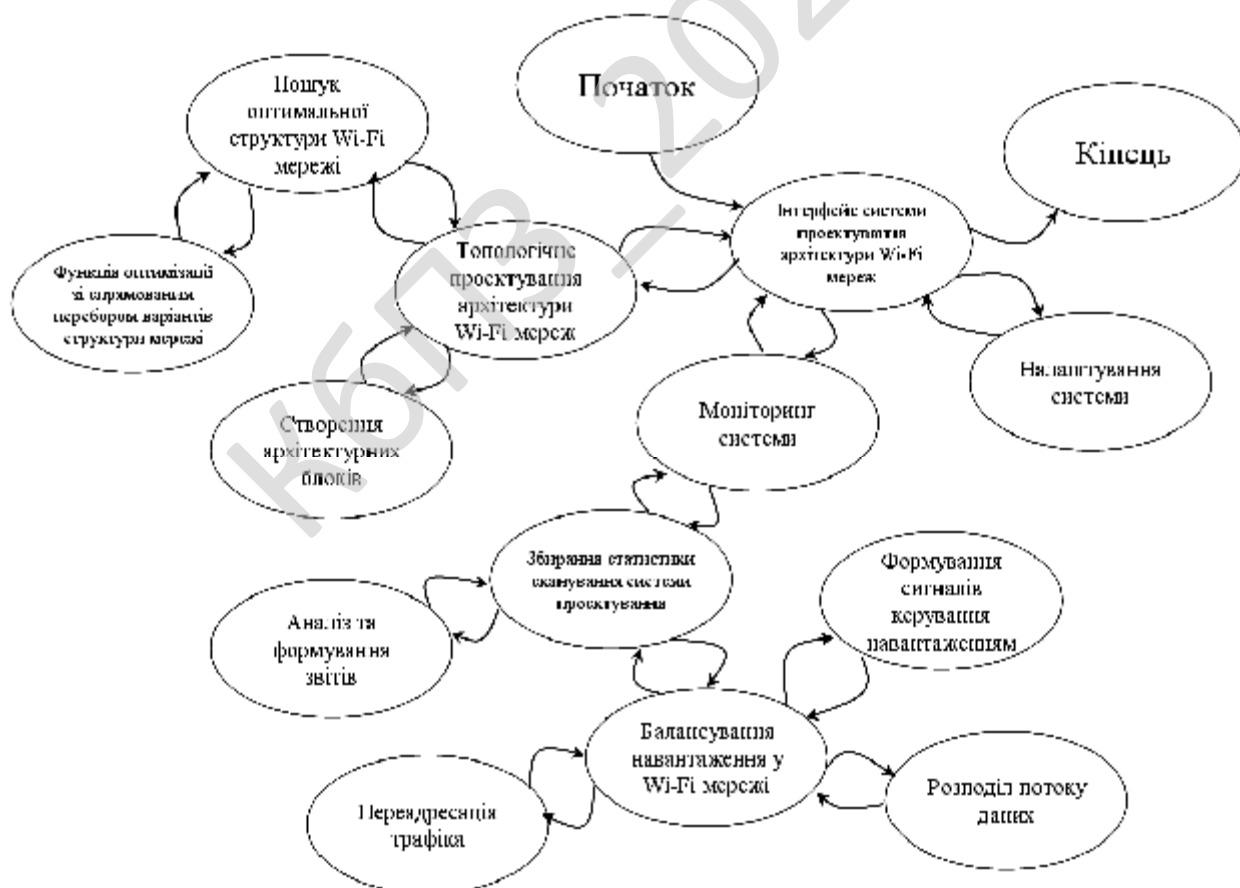


Рисунок 3.3 – Діаграма взаємодії процесів

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Опис функцій ПЗ системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Система проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг є програмним інструментом для автоматизації процесу розрахунку параметрів мережі, моделювання топології та оптимізації використання ресурсів.

Система базується на використанні мови програмування Python та містить набір модулів для аналізу покриття, визначення необхідної кількості точок доступу та оптимізації конфігурації мережі.

Архітектура системи включає наступні модулі:

1. Модуль аналізу покриття обчислює зону покриття кожної точки доступу, враховуючи потужність передавача, втрати сигналу та перешкоди. Він використовує моделі затухання сигналу, такі як модель Neta або модель ITU-R P.1238.

2. Модуль моделювання топології створює оптимальне розташування точок доступу для забезпечення якісного покриття та мінімізації зон без сигналу. Він використовує алгоритми кластеризації, наприклад k-means, для визначення оптимальних місць розміщення точок доступу.

3. Модуль розрахунку продуктивності мережі оцінює максимальну пропускну здатність та середню швидкість підключення користувачів. Він враховує кількість клієнтів на одну точку доступу, смугу пропускання каналу та рівень інтерференції між точками доступу.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 46 |

підходить для використання операторами телекомунікаційних послуг, що планують розгортання бездротових мереж у великих приміщеннях або відкритих просторах.

Частина вихідного коду:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.spatial import distance
from sklearn.cluster import KMeans

# Функція розрахунку покриття точки доступу
def calculate_coverage(tx_power, frequency, environment_factor):

    wavelength = 3e8 / (frequency * 1e9)
    path_loss = 20 * np.log10(wavelength / (4 * np.pi)) + environment_factor
    radius = 10 ** ((tx_power - path_loss) / 20)
    return radius

# Функція оптимального розміщення точок доступу
def optimize_ap_placement(building_map, user_density, n_ap):

    kmeans = KMeans(n_clusters=n_ap, random_state=42)
    kmeans.fit(user_density)
    return kmeans.cluster_centers_

# Функція оцінки продуктивності мережі
def evaluate_network_performance(ap_list, user_list, bandwidth):

    total_users = len(user_list)
    total_aps = len(ap_list)
    avg_bandwidth_per_user = bandwidth / total_users if total_users > 0
else 0

    avg_users_per_ap = total_users / total_aps if total_aps > 0 else 0
    return avg_bandwidth_per_user, avg_users_per_ap

# Функція оптимізації розподілу каналів
def optimize_channel_assignment(ap_list):

    channels = [1, 6, 11]
    assigned_channels = {}
    for i, ap in enumerate(ap_list):
```

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 48 |

```

        assigned_channels[tuple(ap)] = channels[i % len(channels)]
    return assigned_channels

# Функція візуалізації покриття
def generate_coverage_map(ap_list, building_map, radius):

    plt.figure(figsize=(10, 10))
    for ap in ap_list:
        circle = plt.Circle((ap[0], ap[1]), radius, color='b', alpha=0.3)
        plt.gca().add_patch(circle)
        plt.plot(ap[0], ap[1], 'bo')
    plt.xlim(0, np.max(building_map[:, 0]))
    plt.ylim(0, np.max(building_map[:, 1]))
    plt.title("Wi-Fi Coverage Map")
    plt.xlabel("X Coordinate")
    plt.ylabel("Y Coordinate")
    plt.show()

# Приклад використання
building_map = np.random.rand(100, 2) * 100
user_density = np.random.rand(50, 2) * 100
n_ap = 5
tx_power = 20
frequency = 2.4
environment_factor = 40
radius = calculate_coverage(tx_power, frequency, environment_factor)
ap_list = optimize_ap_placement(building_map, user_density, n_ap)
avg_bandwidth_per_user, avg_users_per_ap =
evaluate_network_performance(ap_list, user_density, 100)
assigned_channels = optimize_channel_assignment(ap_list)
generate_coverage_map(ap_list, building_map, radius)

```

Під час роботи над бакалаврською дипломною роботою було створено блок-схеми. Перед їх розглядом необхідно провести роз'яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято

позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). Всередині фігури записують безпосередньо самі операції.

Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елемента. Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елемента. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній). Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 50 |

потім розгалужується. Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи.

Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і передані в нього дані.

Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення). Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об'єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний

функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

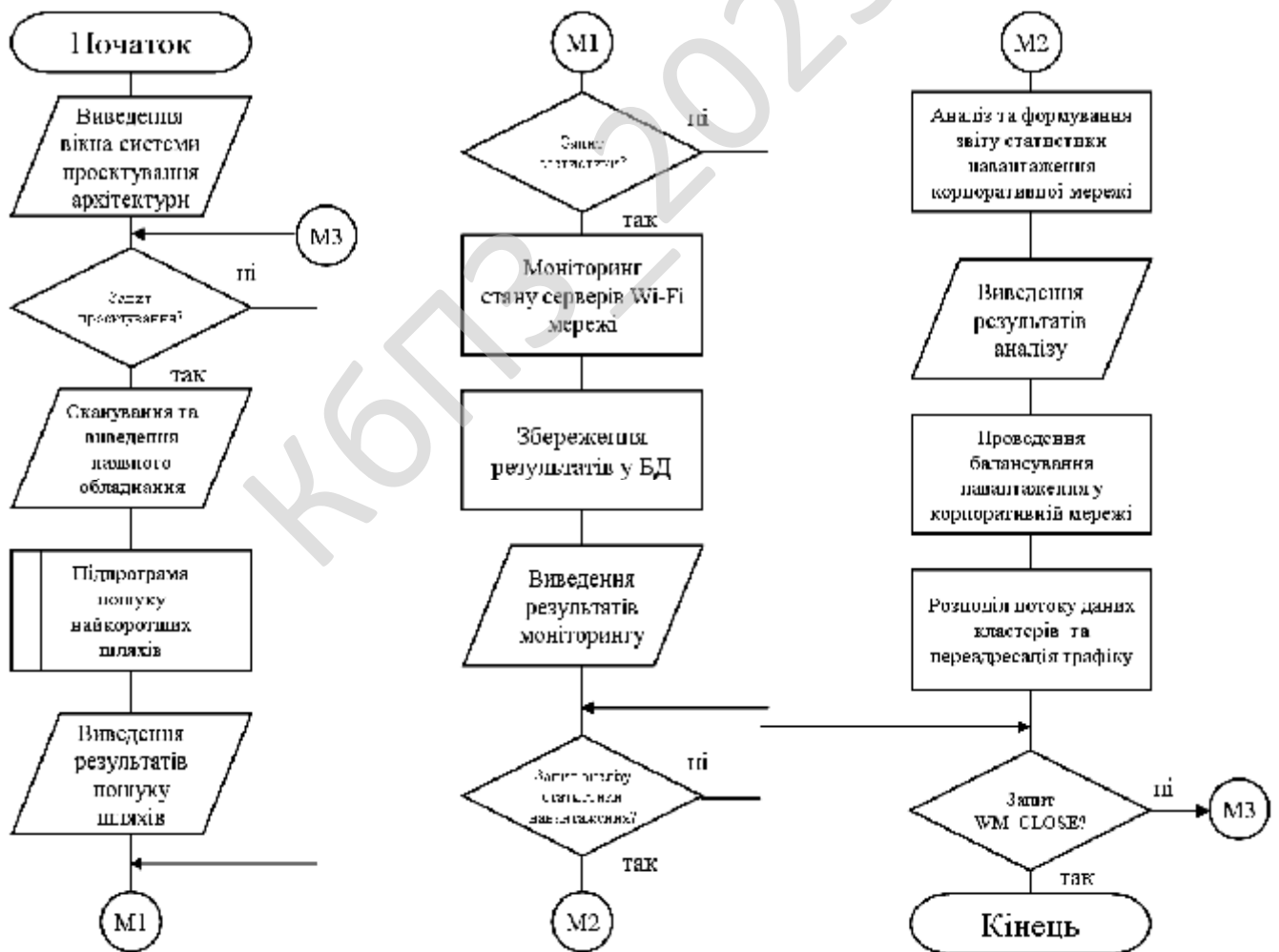


Рисунок 4.1 – Блок-схема основної програми

UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки прикладних програм. Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем.

Діаграми дають можливість представити систему (як ділову, так і програмну) у такому вигляді, щоб її можна було легко перевести в програмний код. Основною причиною використання мови UML є спілкування розробників між собою.

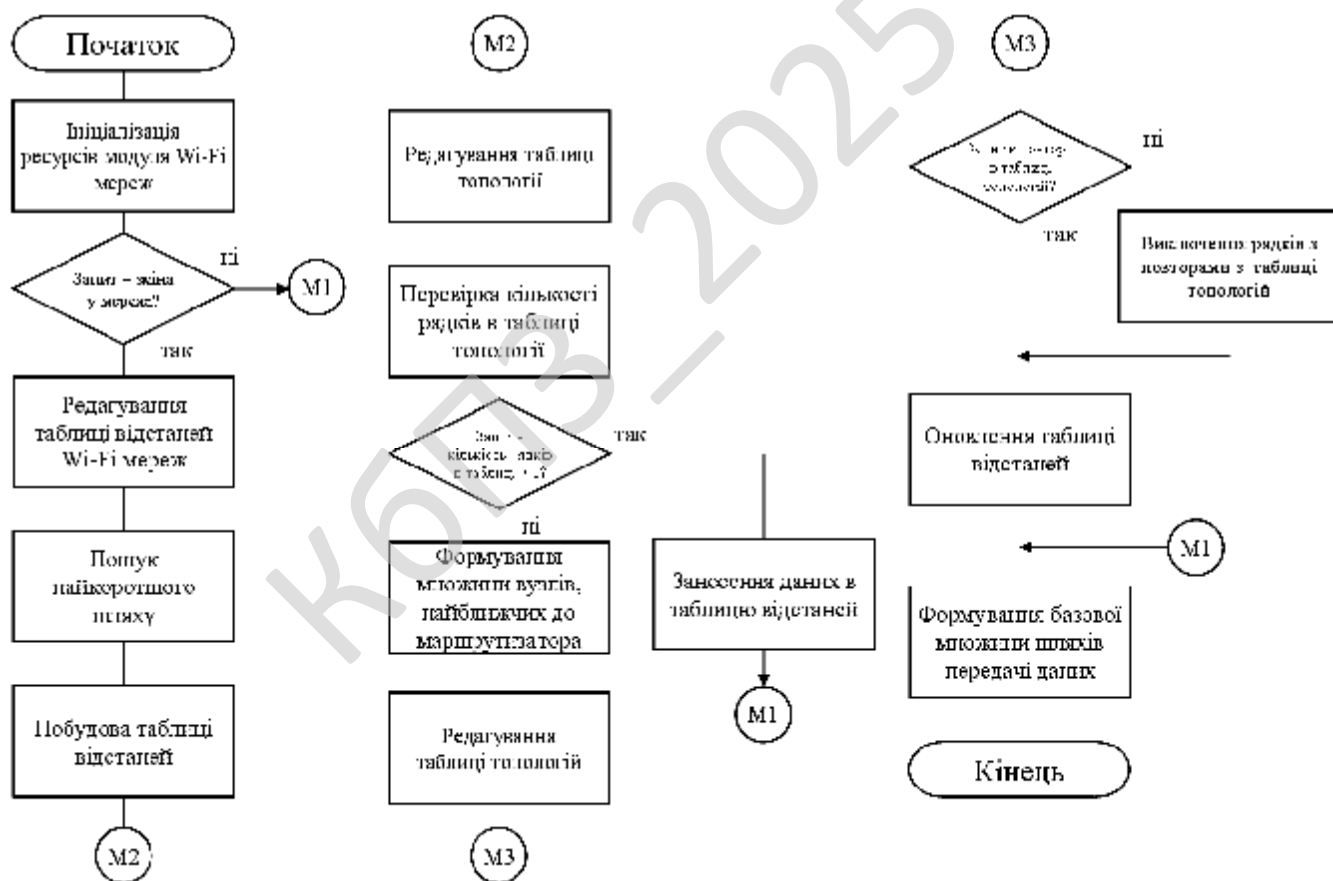


Рисунок 4.2 – Блок-схема роботи підпрограми

Крім того, UML спеціально створювалася для оптимізації процесу розробки програмних систем, що дозволяє збільшити ефективність їх реалізації у кілька разів і помітно поліпшити якість кінцевого продукту.

UML прекрасно зарекомендувала себе в багатьох успішних програмних проектах. Засоби автоматичної генерації кодів дозволяють перетворювати моделі мовою UML у вихідний код об'єктно-орієнтованих мов програмування, що ще більш прискорює процес розробки. Практично усі CASE-засоби (програми автоматизації процесу аналізу і проектування) мають підтримку UML. Моделі розроблені в UML, дозволяють значно спростити процес кодування і направити зусилля програмістів безпосередньо на реалізацію системи.

Діаграми підвищують супроводжуваність проекту і полегшують розробку документації.

UML необхідний:

- Керівникам проектів, які керують розподілом завдань і контролем за проектом.
- Проектувальникам інформаційних систем які розробляють технічні завдання для програмістів.
- Бізнес-аналітикам, які досліджують реальну систему і здійснюють інжиніринг і реінжиніринг бізнесу компанії.
- Програмістам які реалізують модулі інформаційної системи.

При модифікації системи об'єктний підхід дозволяє легко включати в систему нові об'єкти і виключати застарілі без істотної зміни її життєздатності. Використання побудованої моделі при модифікаціях системи дає можливість усунути небажані наслідки змін, оскільки вони не ламають структури системи, а тільки змінюють поведінку об'єктів.

Також при розробці бакалаврської дипломної роботи було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 55 |

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- асоціації (association relationship);
- включення (include relationship);
- розширення (extend relationship);
- узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 56 |

діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *).
Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовани ромбиком.

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм RC5, який являє собою блоковий шифр із безліччю параметрів: розміром блоку, розміром ключа й числом раундів. В алгоритмі RC5 передбачені три операції: XOR, додавання й циклічні зрушення. На більшості процесорів операції циклічного зрушення виконуються за постійний час, змінні

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 58 |

циклічні зрушення являють собою нелінійну функцію. Циклічні зрушення залежать як від ключа, так і від даних.

В RC5 використовується блок змінної довжини, але в приводиться прикладі, що буде розглянутий, 64-бітовий блок даних. Шифрування використовує $2r+2$ залежних від ключа 32-бітових слів – $S_0, S_1, S_2, \dots, S_{2r+1}$ – де r – число раундів. Для шифрування спочатку потрібно розділити блок відкритого тексту на два 32-бітових слова: A і B . (При впакуванні байтів у слова в алгоритмі RC5 дотримується угода про прямий порядок (little-endian) байтів: перший байт займає молодші біти регістра A й т. ін.) Потім:

$$A = A + S_0$$

$$B = B + S_0$$

Для i від 1 до r :

$$A = ((A \oplus B) \lll B) + S_{2i}$$

$$B = ((B \oplus A) \lll A) + S_{2i+1}$$

Вихід перебуває в регістрах A і B .

Розшифрування теж нескладно. Потрібно розбити блок відкритого тексту на два слова, A й B , а потім:

Для i від r до 1 із кроком -1:

$$B = ((B - S_{2i+1}) \ggg A) \oplus A$$

$$A = ((A - S_{2i}) \ggg B) \oplus B$$

$$B = B - S_i$$

$$A = A - S_0$$

Символом « \ggg » позначене циклічне зрушення вправо. Звичайно ж, всі додавання й вирахування виконуються по модулю 2^{32} .

Створення масиву ключів складніше, але теж прямолінійно. Спочатку байти ключа копіюються в масив L із 32-бітових слів, доповнюючи при необхідності заключне слово нулями. Потім масив S ініціалізується за допомогою лінійного конгруентного генератора по модулю 2^{32} :

$$S_0 = P$$

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 59 |

Для i від 1 до $2(r + 1) - 1$:

$$S_i = (S_{i-1} + Q) \bmod 2^{32}$$

де $P = 0xb7e15163$ і $Q = 0x9e3779b9$.

Нарешті, потрібно підставити L в S :

$$i = j = 0$$

$$A = B = 0$$

Виконати $3n$ раз (де n – максимум від $2(r + 1)$ і c):

$$A = S_i = (S_i + A + B) \lll 3$$

$$B = L_i = (L_i + A + B) \lll (A + B)$$

$$i = (i + 1) \bmod 2(r + 1)$$

$$j = (j + 1) \bmod c$$

КБПЗ - 2025

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 60 |

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг яке зображено на рисунку 5.1. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

- Навігаційне меню: Файл; Налаштування; Статистика сканування; Довідка.
- Функції представлені у графічному вигляді.
- Розділу виведення результату роботи системи.
- Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.
- Функціональних кнопок ПЗ.

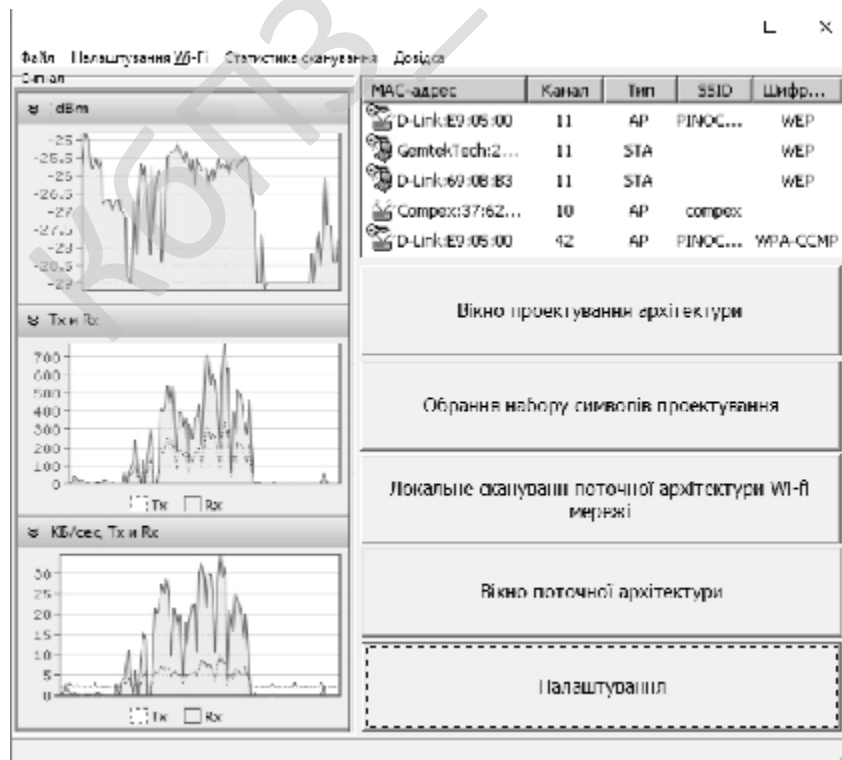


Рисунок 5.1 – Головне вікно ПЗ

Wi-Fi стає усе популярніше як технологія для голосового зв'язку, альтернативна класичним стільниковим системам. Тому одним з важливих вимог до сучасних рішень Wi-Fi, поряд з високою продуктивністю і ємністю, є безшовний хендовер (стандарт IEEE 802.11r) – швидке перемикання клієнта на іншу точку доступу без перерви сервісу.

Крім того, замовники зацікавлені в тому, щоб при розгортанні мереж Wi-Fi «поменше копати й тягти кабелі». Задовольнити ця вимога, як вважає фахівець Cambium Networks, допомагає підтримка ніздрюватих топологій (mesh), а також наявність доступних за ціною підключень для точок доступу (backhaul). Зв'язок із вищестоящими вузлами може бути забезпечена, зокрема, за рахунок ресурсів самої ТД – наприклад, у діапазоні 5 ГГц. Для спрощення й прискорення розгортання мережі також важлива наявність засобів автоконфігурування й шаблонів для налаштування.

Керований хот-спот найчастіше можна виявити в ресторанах і кафе, на вокзалах, станціях метро, в аеропортах, у навчальних закладах і кампусах.

Одержав поширення також Wi-Fi на території лікарень, міських площ, парків і стадіонів.

Багато компаній організують хот-споти у своїх офісах для зручності клієнтів.

У ряді випадків Wi-Fi може виявитися єдиним способом організувати доступ до Мережі, наприклад, якщо підприємство перебуває в зоні з поганим покриттям стільникових мереж.

Wi-Fi для кафе й ресторанів дозволяє залучити додаткових клієнтів, хот-спот для готелів забезпечує закладу приплив відвідувачів із числа ділових людей і активних учасників соціальних мереж.

Wi-Fi для торгових центрів дає можливість збільшити кількість покупців, а хот-спот для лікарень і поліклінік дозволяє підняти рівень обслуговування на новий щабель.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 62 |

Розроблена програма має дуже простий і інтуїтивно зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

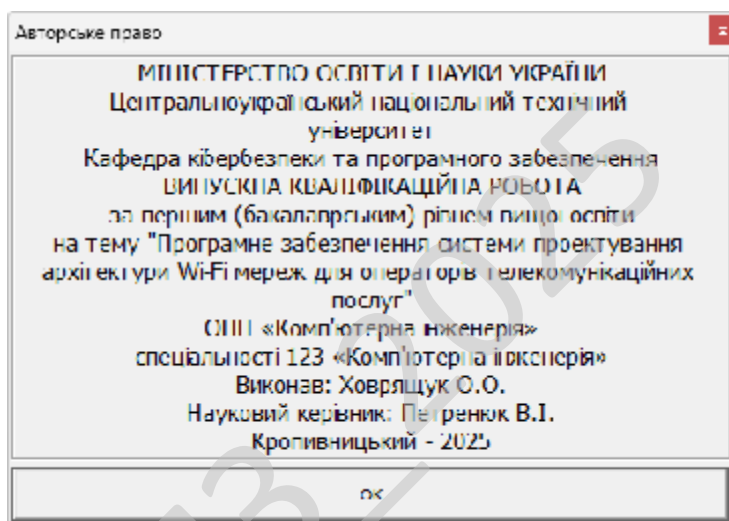


Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є

унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

- Випуск.
- Встановлення та активація.
- Деактивація.
- Адаптація.
- Обновлення.
- Вмонтування.
- Відстежування версій.
- Видалення.
- Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження.

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 64 |

в IT рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

- відповідності поставленим вимогам;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

- Гарантується перевірка всіх незалежних маршрутів програми.
- Знаходяться гілки True, False для всіх логічних рішень.
- Виконуються всі цикли (у межах їхніх кордонів та діапазонів).
- Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 65 |

- Кількість незалежних маршрутів може бути дуже велика.
- Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.
- У програмі можуть бути пропущені деякі маршрути.
- Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

- Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.
- Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.
- При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Обрано умови розповсюдження – Freeware.

Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільноті як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

Безплатне програмне забезпечення можна безоплатно встановлювати та використовувати (іноді з певними обмеженнями, як, наприклад, «безплатне для домашнього або некомерційного вжитку»), в той час як вільне програмне забезпечення можна продавати за будь-яку суму, але при тому, у користувача, котрий його отримує, повинні бути права на вивчення, модифікацію та поширення сирцевих кодів одержаної програми.

КБПЗ_2025

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 67 |

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

Рішення завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

– Досліджена система проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

– На основі отриманих результатів досліджень створена програмна реалізація системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 68 |

системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм RC5.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 69 |

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Оліфер В.Г. Комп'ютерні мережі. Принципи, технології, протоколи. Підручник / В.Г. Оліфер, Н.А.Оліфер. – [5-е вид.]. – 2016. – 944 с.
2. Е. Таненбаум, Д. Уезеролл «Комп'ютерні мережі». – [5-е вид.]. – 2016. – 960 с.
3. Wendell Odom. «CCNA 200-301 Official Cert Guide, Volume 1». Cisco Press. 2020. – 848 p.
4. Wendell Odom. «CCNA 200-301 Official Cert Guide, Volume 2 Premium Edition eBook and Practice Test». Cisco Press. 2020. – 624 p.
5. Scott Jernigan «CompTIA Network+ Certification All-in-One Exam Guide, Eighth Edition». 2022. – 976 p.
6. Doug Lowe «Networking For Dummies 12th Edition». 2020. – 480 p.
7. Ramon Nastase «Computer Networking: The Beginner's guide for Mastering Computer Networking, the Internet and the OSI Model». 2018. – 186 p.
8. Russ White & Ethan Banks «Computer Networking Problems and Solutions: An Innovative Approach to Building Resilient, Modern Networks». 2017. – 832 p.
9. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.
10. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.
11. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yanchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 70 |

12. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.
13. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». *CEUR Workshop Proceedings Volume 3156*, 2022, Pages 390-399.
14. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». *Communications in Computer and Information Science*, 2021, vol 1486. Springer, Cham. pp 169-184.
15. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.
16. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.
17. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 172-177.
18. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) *Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies*, vol 48. Springer, Cham. 2021. pp 557-587.

19. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

20. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 366-379.

21. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 633-645.

22. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». *International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019*; Odessa; Ukraine; 9-13 September 2019. P.22-28.

23. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». *International Journal of Computing*; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

24. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

25. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyz, M., «QoE optimization technique for media delivery in 5G networks». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.

26. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.

27. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», *2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019*, P. 395-399.

28. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 353-358.

29. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 347-352.

30. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», *CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019*, Pages 618-629.

31. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», *Telecommunications and Radio Engineering*. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

32. Смірнова Т.В., Коноплицька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка*. 2024. №4(24), С. 6-27.

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 73 |

33. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології*, 2024, № 13, с. 28-35.

34. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.

35. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління*, № 2(70). 2022. С. 28-37.

36. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 3(69). С. 93-98.

37. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»*, № 2 (307). С. 46-52. 2022.

38. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

39. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 74 |

процесу відновлення поверхонь конструкцій і деталей машин». *Сучасні інформаційні системи*. 2021. Т. 5, № 4. С. 79-95

40. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» *Вісник Черкаського державного технологічного університету. Технічні науки*. №4. С. 103-110. 2020.

41. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», *Кібербезпека: освіта, наука, техніка*. № 3(7). С. 43-62. 2020.

42. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В., Поліщук Л.І. Інформаційна безпека в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2020. – 294 с.

43. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у *Кібербезпека та інформаційні технології: монографія*. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

44. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». *Центральноукраїнський науковий вісник. Технічні науки*. № 2(33). с. 161-172, 2019.

45. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

46. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 75 |

47. Смірнов О.А., Дреєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

48. Дреєва Г.М., Смірнов О.А., Дреєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

49. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

50. Смірнов О.А., Смірнов С.А., Поліщук Л.І., Смірнова Т.В., Коноплицька-Слободенюк О.К. Метод формування антивірусного захисту даних з використанням безпечної маршрутизації метаданих. Кібербезпека: освіта, наука, техніка. – Том 3 № 3. – Київ: КУ ім. Бориса Грінченка. – 2019. – С. 63-87.

51. Смірнов О.А., Гнатюк С.О., Кавун С.В., Терейковський І.А., Жмурко Т.О., Смірнов С.А., Коваленко А.С. Основи безпеки в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2018. – 177 с.

52. Смірнов О.А., Котелянець В.В. Стійкі до колізій стохастичні моделі функціонування безпроводових сенсорних мереж. Вісник інженерної академії України, №3, с. 145-152, 2018

| | | | | | | |
|------|------|----------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ПЗ | Арк. |
| Вим. | Арк. | № докум. | Підпис | Дата | | 76 |

Додаток А
(обов'язковий)

Технічне завдання

Зміст

| | | |
|-------|---|---|
| 1 | Найменування та область застосування..... | 2 |
| 2 | Підстава для розробки..... | 2 |
| 3 | Мета та призначення розробки..... | 2 |
| 4 | Джерела розробки..... | 2 |
| 5 | Технічні вимоги..... | 2 |
| 5.1 | Вміст проекту..... | 2 |
| 5.2 | Показники призначення..... | 3 |
| 5.3 | Вимоги до функціональних характеристик..... | 3 |
| 5.4 | Вимоги до архітектури..... | 3 |
| 5.5 | Вимоги до надійності..... | 3 |
| 5.6 | Умови експлуатації..... | 4 |
| 5.7 | Вимоги до складу та параметрів технічних засобів..... | 4 |
| 5.8 | Вимоги до інформаційної і програмної сумісності..... | 4 |
| 5.8.1 | Обладнання..... | 4 |
| 5.8.2 | Мова програмування..... | 4 |
| 5.8.3 | Вхідні дані..... | 5 |
| 5.8.4 | Вихідні дані..... | 5 |
| 6 | Вимоги до програмної документації..... | 5 |
| 7 | Перелік документів, що розробляються..... | 5 |
| 8 | Етапи розробки..... | 6 |
| 9 | Порядок контролю та приймання..... | 6 |

| | | | | | | | | |
|------------------|-----------------------|--------------------|---------------|-------------|--|-------------|--------------|----------------|
| | | | | | ВКРБ-123.25.0044.00.00.ТЗ | | | |
| <i>Вим.</i> | <i>Арк.</i> | <i>№ документа</i> | <i>Підпис</i> | <i>Дата</i> | | | | |
| <i>Розробив</i> | <i>Ховряцук О.О.</i> | | | | <i>Програмне забезпечення системи проекткування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг</i> | <i>Літ.</i> | <i>Аркуш</i> | <i>Аркушів</i> |
| <i>Перевірів</i> | <i>Петренюк В.І.</i> | | | | | <i>Б</i> | <i>1</i> | <i>6</i> |
| <i>Н. Контр.</i> | <i>Коваленко А.С.</i> | | | | <i>ЦНТУ КІ-21-2</i> | | | |
| <i>Затв.</i> | <i>Смірнов О.А.</i> | | | | | | | |

1 Найменування та область застосування

Це технічне завдання розповсюджується на розробку системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 47-02 від 17.01.2025 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти є розробка програмного забезпечення системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;

| | | | | | | |
|------|------|-------------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ТЗ | Арк. |
| Вим. | Арк. | № документа | Підпис | Дата | | 2 |

- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- системи проектування архітектури Wi-Fi мереж для операторів телекомунікаційних послуг;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

| | | | | | | |
|------|------|-------------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ТЗ | Арк. |
| Вим. | Арк. | № документа | Підпис | Дата | | 3 |

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище Python.

| | | | | | | |
|------|------|-------------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ТЗ | Арк. |
| Вим. | Арк. | № документа | Підпис | Дата | | 2 |

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Перелік документів, що розробляються

- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Пояснювальна записка – 76 аркушів.

8 Етапи розробки

8.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти (складання ТЗ).

| | | | | | | |
|------|------|-------------|--------|------|----------------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ТЗ | Арк. |
| Вим. | Арк. | № документа | Підпис | Дата | | 5 |

8.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.

8.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

8.4 Побудова схем взаємодії даних.

8.5 Створення прототипу ПЗ.

8.6 Віднаходження ПЗ, аналіз отриманих результатів.

8.7 Оформлення пояснювальної записки і виконання робіт по графічній частині.

9 Порядок контролю та приймання

9.1 Подання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти на попередній захист 23.05.2025 р.

9.2 Подання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти на захист 3.06.2025 р.

| | | | | | | |
|------|------|-------------|--------|------|---------------------------|------|
| | | | | | ВКРБ-123.25.0044.00.00.ТЗ | Арк. |
| Вим. | Арк. | № документа | Підпис | Дата | | 6 |

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
першим (бакалаврським) рівнем вищої освіти

_____ Петренюк В.І.

*Програмне забезпечення системи проектування архітектури Wi-Fi мереж
для операторів телекомунікаційних послуг*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 19

Літера: РП

Кропивницький – 2025 року

Основна програма

```
import random
import time
import threading
import json

# Клас для VLAN-менеджменту
class VLANManager:

    def __init__(self):
# Ініціалізація VLAN
        self.vlans = {}

    def create_vlan(self, vlan_id, name):
# Створення VLAN
        if vlan_id not in self.vlans:
            self.vlans[vlan_id] = {"name": name, "devices": []}

    def assign_device(self, vlan_id, device_mac):
# Додавання пристрою у VLAN
        if vlan_id in self.vlans:
            self.vlans[vlan_id]["devices"].append(device_mac)

# Клас керування точками доступу
class AccessPoint:

    def __init__(self, name, location, capacity):
# Ініціалізація точки доступу
        self.name = name
        self.location = location
        self.capacity = capacity
        self.connected_devices = []

    def connect_device(self, device):
# Підключення пристрою
        if len(self.connected_devices) < self.capacity:
            self.connected_devices.append(device)

# Клас AI-балансування навантаження
class AILoadBalancer:

    def __init__(self, access_points):
# Ініціалізація балансування
        self.access_points = access_points

    def distribute_load(self, devices):
# Розподіл пристроїв
        for device in devices:
            best_ap = min(self.access_points, key=lambda ap:
len(ap.connected_devices))
            best_ap.connect_device(device)

# Клас моніторингу трафіку
class TrafficMonitor:

    def __init__(self):
# Ініціалізація моніторингу
        self.traffic_data = []

    def log_traffic(self, ap_name, data_used):
# Логування трафіку
```

```

        self.traffic_data.append({"ap_name": ap_name, "data_used": data_used,
"timestamp": time.time()})

    def get_total_traffic(self):
# Отримання загального трафіку
        return sum(entry["data_used"] for entry in self.traffic_data)

# Клас прогнозування навантаження
class TrafficForecaster:

    def __init__(self):
# Ініціалізація історичних даних
        self.traffic_data = []

    def log_traffic(self, timestamp, traffic_amount):
# Логування використання трафіку
        self.traffic_data.append({"timestamp": timestamp, "traffic":
traffic_amount})

    def forecast_traffic(self, next_hours=1):
# Прогнозування трафіку
        if not self.traffic_data:
            return "Недостатньо даних"
        avg_traffic = sum(entry["traffic"] for entry in self.traffic_data) /
len(self.traffic_data)
        return avg_traffic * next_hours

# Клас керування IoT-пристроями
class IoTManager:

    def __init__(self):
# Ініціалізація IoT
        self.devices = {}

    def register_device(self, device_id, device_type):
# Реєстрація пристрою
        self.devices[device_id] = {"type": device_type, "firmware_version":
"1.0"}

    def update_firmware(self, device_id):
# Оновлення прошивки IoT
        if device_id in self.devices:
            self.devices[device_id]["firmware_version"] = "1.1"

# Основний клас мережі Wi-Fi
class WiFiNetwork:

    def __init__(self):
# Ініціалізація точок доступу
        self.access_points = []
        self.vlan_manager = VLANManager()
        self.traffic_monitor = TrafficMonitor()
        self.forecaster = TrafficForecaster()
        self.iot_manager = IoTManager()

    def add_access_point(self, ap_name, location, capacity):
# Додавання точки доступу
        ap = AccessPoint(ap_name, location, capacity)
        self.access_points.append(ap)

    def connect_user(self, user):
# Підключення користувача

```

```

        best_ap = min(self.access_points, key=lambda ap:
len(ap.connected_devices))
        best_ap.connect_device(user)

    def monitor_network(self):
# Моніторинг мережі
        for ap in self.access_points:
            data_used = random.randint(1, 100)
            self.traffic_monitor.log_traffic(ap.name, data_used)

    def get_total_traffic(self):
# Отримання загального трафіку
        return self.traffic_monitor.get_total_traffic()

# Ініціалізація мережі
wifi_network = WiFiNetwork()

# Додавання точок доступу
wifi_network.add_access_point("AP1", (0, 0), 10)
wifi_network.add_access_point("AP2", (10, 10), 15)

# Створення користувачів
users = [f"user{i}" for i in range(20)]

# Підключення користувачів до мережі
for user in users:
    wifi_network.connect_user(user)

# Моніторинг трафіку у фоновому режимі
def traffic_monitoring():
    while True:
        wifi_network.monitor_network()
        time.sleep(5)

monitoring_thread = threading.Thread(target=traffic_monitoring, daemon=True)
monitoring_thread.start()

# Симуляція роботи мережі
time.sleep(30)

# Виведення загального трафіку
print(f"Загальний трафік використаний: {wifi_network.get_total_traffic()} МВ")

```

Файл `vlan_management.py`

```
# Розширене керування VLAN: збереження в файл, логування змін, імпорт/експорт

import json

# Клас для керування VLAN у Wi-Fi мережі
class VLANManager:

    def __init__(self, config_file="vlans.json"):

# Ініціалізація словника VLAN та файлу конфігурації
        self.vlans = {}
        self.config_file = config_file
        self.load_from_file()

    def create_vlan(self, vlan_id, name):

# Створення нового VLAN
        if vlan_id not in self.vlans:
            self.vlans[vlan_id] = {"name": name, "devices": []}
            self.save_to_file()
            print(f"VLAN {vlan_id} - {name} створено.")
            return True

        print(f"Помилка: VLAN {vlan_id} вже існує.")
        return False

    def delete_vlan(self, vlan_id):

# Видалення VLAN
        if vlan_id in self.vlans:
            del self.vlans[vlan_id]
            self.save_to_file()
            print(f"VLAN {vlan_id} видалено.")
            return True

        print(f"Помилка: VLAN {vlan_id} не знайдено.")
        return False

    def assign_device_to_vlan(self, vlan_id, device_mac):

# Призначення пристрою до VLAN
        if vlan_id in self.vlans:
            self.vlans[vlan_id]["devices"].append(device_mac)
            self.save_to_file()
            return True
        return False

    def save_to_file(self):

# Збереження VLAN у файл
        with open(self.config_file, "w") as file:
            json.dump(self.vlans, file, indent=4)

    def load_from_file(self):

# Завантаження VLAN із файлу
        try:
            with open(self.config_file, "r") as file:
                self.vlans = json.load(file)
        except FileNotFoundError:
```

```
self.vlans = {}

# Тестування VLANManager
if __name__ == "__main__":
    vlan_manager = VLANManager()
    vlan_manager.create_vlan(10, "Guest")
    vlan_manager.create_vlan(20, "Corporate")

    vlan_manager.assign_device_to_vlan(10, "AA:BB:CC:DD:EE:FF")
    vlan_manager.assign_device_to_vlan(20, "11:22:33:44:55:66")
    print("VLAN List:", vlan_manager.vlans)
```

КБПЗ_2025

Файл handover_manager.py

```
import random
import time

# Клас для автоматичного перемикування між точками доступу
class HandoverManager:

    def __init__(self, access_points):
# Ініціалізація списку точок доступу
        self.access_points = access_points

    def get_best_access_point(self, device_location):
# Пошук найкращої точки доступу на основі сигналу та відстані
        best_ap = min(self.access_points, key=lambda ap:
self._calculate_signal_strength(ap["location"], device_location))
        return best_ap

    def _calculate_signal_strength(self, ap_location, device_location):
# Розрахунок сили сигналу
        distance = ((ap_location[0] - device_location[0]) ** 2 + (ap_location[1]
- device_location[1]) ** 2) ** 0.5
        signal_strength = max(100 - distance * 10, 0) # Умовна формула
зменшення сигналу
        return signal_strength

# Тестування HandoverManager
if __name__ == "__main__":
    access_points = [{"name": "AP1", "location": (0, 0)}, {"name": "AP2",
"location": (10, 10)}, {"name": "AP3", "location": (5, 5)}]
    handover_manager = HandoverManager(access_points)

    for _ in range(5):
        device_location = (random.randint(0, 10), random.randint(0, 10))
        best_ap = handover_manager.get_best_access_point(device_location)
        print(f"Device at {device_location} connects to {best_ap['name']}")
        time.sleep(2)
```

Файл ai_load_balancer.py

```
import random

# Балансування навантаження з використанням AI-алгоритмів
class AILoadBalancer:

    def __init__(self, access_points):
# Ініціалізація точок доступу
        self.access_points = access_points

    def distribute_load(self, devices):
# Розподіл навантаження між точками доступу
        for device in devices:
            best_ap = self._select_best_access_point()
            best_ap["connected_devices"].append(device)
            print(f"Device {device} connected to {best_ap['name']}")

    def _select_best_access_point(self):
# Вибір найменш завантаженої точки доступу
        return min(self.access_points, key=lambda ap:
len(ap["connected_devices"]))

# Тестування AILoadBalancer
if __name__ == "__main__":
    access_points = [{"name": "AP1", "connected_devices": []}, {"name": "AP2",
"connected_devices": []}, {"name": "AP3", "connected_devices": []}]
    ai_balancer = AILoadBalancer(access_points)

    devices = [f"Device_{i}" for i in range(10)]
    ai_balancer.distribute_load(devices)

    for ap in access_points:
        print(f"{ap['name']} has {len(ap['connected_devices'])} devices
connected")
```

Файл radius_authentication.py

```
import hashlib

# Імітація RADIUS-сервера для аутентифікації користувачів
class RadiusServer:

    def __init__(self):
# Ініціалізація списку користувачів
        self.users = {}
        self.failed_attempts = {}

    def add_user(self, username, password):
# Додавання користувача у систему
        password_hash = hashlib.sha256(password.encode()).hexdigest()
        self.users[username] = password_hash

    def authenticate(self, username, password):
# Аутентифікація користувача
        password_hash = hashlib.sha256(password.encode()).hexdigest()
        if username not in self.users:
            return "Користувач не знайдений"

        if self.users.get(username) == password_hash:
            self.failed_attempts[username] = 0
            return "Аутентифікація успішна"

        self.failed_attempts[username] = self.failed_attempts.get(username, 0) +
1
        if self.failed_attempts[username] >= 3:
            return "Обліковий запис тимчасово заблоковано"

        return "Невірний пароль"

# Тестування RadiusServer
if __name__ == "__main__":
    radius = RadiusServer()
    radius.add_user("admin", "securepassword")

    print(radius.authenticate("admin", "wrongpassword"))
    print(radius.authenticate("admin", "wrongpassword"))
    print(radius.authenticate("admin", "wrongpassword"))
    print(radius.authenticate("admin", "securepassword"))
```

Файл network_visualization.py

```
import random
import time
import matplotlib.pyplot as plt

# Клас для візуалізації покриття Wi-Fi мережі
class NetworkVisualizer:

    def __init__(self, access_points):
# Ініціалізація точок доступу
        self.access_points = access_points
        self.devices = []

    def add_device(self, x, y):
# Додавання пристрою до візуалізації
        device = {"x": x, "y": y}
        self.devices.append(device)
        print(f"Додано пристрій на координати ({x}, {y})")

    def update_visualization(self):
# Оновлення графічної візуалізації мережі
        plt.figure(figsize=(8, 8))
        for ap in self.access_points:
            plt.scatter(ap["x"], ap["y"], c='red', label=ap["name"], s=200)
            plt.text(ap["x"] + 0.2, ap["y"] + 0.2, ap["name"], fontsize=12,
color='black')

        for device in self.devices:
            plt.scatter(device["x"], device["y"], c='blue', s=100)

        plt.xlabel("X координата")
        plt.ylabel("Y координата")
        plt.title("Wi-Fi Покриття Мережі")
        plt.grid()
        plt.show()

# Тестування NetworkVisualizer
if __name__ == "__main__":
    access_points = [{"name": "AP1", "x": 0, "y": 0}, {"name": "AP2", "x": 10,
"y": 10}]
    visualizer = NetworkVisualizer(access_points)

    for _ in range(5):
        x, y = random.randint(0, 15), random.randint(0, 15)
        visualizer.add_device(x, y)

    visualizer.update_visualization()
```

Файл traffic_forecasting.py

```
import random
import time

# Клас прогнозування навантаження на мережу
class TrafficForecaster:

    def __init__(self):
# Ініціалізація історичних даних трафіку
        self.traffic_data = []

    def log_traffic(self, timestamp, traffic_amount):
# Логування використання трафіку
        self.traffic_data.append({"timestamp": timestamp, "traffic":
traffic_amount})

    def forecast_traffic(self, next_hours=1):
# Простий алгоритм прогнозування (середнє значення трафіку)
        if len(self.traffic_data) == 0:
            return "Недостатньо даних для прогнозування."

        avg_traffic = sum([entry["traffic"] for entry in self.traffic_data]) /
len(self.traffic_data)
        predicted_traffic = avg_traffic * next_hours
        return f"Прогнозоване використання трафіку на {next_hours} год.:
{predicted_traffic:.2f} МБ"

# Тестування TrafficForecaster
if __name__ == "__main__":
    forecaster = TrafficForecaster()

    for _ in range(10):
        timestamp = time.time()
        traffic_amount = random.randint(50, 500)
        forecaster.log_traffic(timestamp, traffic_amount)

    print(forecaster.forecast_traffic(5))
```

Файл ddos_detection.py

```
import random
import time

# Клас для виявлення DDoS-атак
class DDoSDetector:

    def __init__(self, threshold):
# Ініціалізація порогового значення атак
        self.threshold = threshold
        self.traffic_counts = {}
        self.blacklist = set()

    def detect_attack(self, ip_address):
# Виявлення аномальної активності
        if ip_address in self.blacklist:
            print(f"⚠ IP {ip_address} заблоковано через попередні атаки!")
            return True

        if ip_address not in self.traffic_counts:
            self.traffic_counts[ip_address] = 1
        else:
            self.traffic_counts[ip_address] += 1

        if self.traffic_counts[ip_address] > self.threshold:
            print(f"⚠ Попередження: DDoS-атака з IP {ip_address}!
Блокування...")
            self.blacklist.add(ip_address)
            return True

        return False

# Тестування DDoSDetector
if __name__ == "__main__":
    detector = DDoSDetector(threshold=5)

    test_ips = ["192.168.1.1", "192.168.1.2", "192.168.1.3"]

    for _ in range(20):
        ip = random.choice(test_ips)
        detector.detect_attack(ip)
        time.sleep(0.5)
```

Файл guest_network.py

```
import time

# Клас для створення гостьових Wi-Fi мереж
class GuestNetworkManager:

    def __init__(self):
# Ініціалізація активних гостьових мереж
        self.guest_networks = {}

    def create_guest_network(self, ssid, duration):
# Створення нової гостьової мережі з обмеженим часом
        expiration_time = time.time() + duration
        self.guest_networks[ssid] = expiration_time
        print(f"Гостьова мережа {ssid} активна на {duration} секунд.")

    def check_networks(self):
# Перевірка активних мереж
        current_time = time.time()
        for ssid in list(self.guest_networks.keys()):
            if current_time > self.guest_networks[ssid]:
                print(f"Гостьова мережа {ssid} вимкнена.")
                del self.guest_networks[ssid]

# Тестування GuestNetworkManager
if __name__ == "__main__":
    guest_manager = GuestNetworkManager()

    guest_manager.create_guest_network("Guest_WiFi_1", 10)
    guest_manager.create_guest_network("Guest_WiFi_2", 15)

    time.sleep(12)
    guest_manager.check_networks()
```

```
import random

# Клас для генерації IPv6-адрес
class IPv6Manager:

    def __init__(self):
# Ініціалізація списку адрес
        self.allocated_addresses = set()

    def generate_ipv6_address(self):
# Генерація випадкової IPv6-адреси
        ipv6_address = ":".join(f"{random.randint(0, 65535):x}" for _ in
range(8))
        if ipv6_address not in self.allocated_addresses:
            self.allocated_addresses.add(ipv6_address)
            return ipv6_address
        return self.generate_ipv6_address()

    def list_allocated_addresses(self):
# Список виділених IPv6-адрес
        return list(self.allocated_addresses)

# Тестування IPv6Manager
if __name__ == "__main__":
    ipv6_manager = IPv6Manager()

    for _ in range(5):
        print(f"Виділена IPv6-адреса: {ipv6_manager.generate_ipv6_address()}")

    print("Усі виділені адреси:", ipv6_manager.list_allocated_addresses())
```

Файл event_logging.py

```
import sqlite3
import time

# Клас для логування подій у базу даних SQLite
class EventLogger:

    def __init__(self, db_name="events.db"):
# Ініціалізація підключення до бази даних
        self.conn = sqlite3.connect(db_name)
        self.cursor = self.conn.cursor()
        self._create_table()

    def _create_table(self):
# Створення таблиці для логування, якщо вона не існує
        self.cursor.execute("""
            CREATE TABLE IF NOT EXISTS logs (
                id INTEGER PRIMARY KEY AUTOINCREMENT,
                timestamp TEXT,
                event_type TEXT,
                level TEXT,
                details TEXT
            )
        """)
        self.conn.commit()

    def log_event(self, event_type, level, details):
# Додавання нового запису у лог
        timestamp = time.strftime("%Y-%m-%d %H:%M:%S")
        self.cursor.execute("INSERT INTO logs (timestamp, event_type, level,
details) VALUES (?, ?, ?, ?)",
            (timestamp, event_type, level, details))

        self.conn.commit()
        print(f"Подія '{event_type}' [{level}] залогована: {details}")

    def fetch_logs(self):
# Отримання всіх записів з логів
        self.cursor.execute("SELECT * FROM logs")
        return self.cursor.fetchall()

# Тестування EventLogger
if __name__ == "__main__":
    logger = EventLogger()
    logger.log_event("Network_Error", "ERROR", "Втрачено з'єднання з точкою
доступу AP1")
    logger.log_event("Authentication", "INFO", "Користувач admin увійшов у
систему")
    print("Всі логовані події:", logger.fetch_logs())
```

Файл m_integration.py

```
import json

# Клас для керування API-запитами мобільного додатка
class MobileAPI:

    def __init__(self):
# Ініціалізація списку користувачів мобільного додатка
        self.users = {}
        self.notifications = {}

    def register_user(self, username, device_id):
# Реєстрація користувача у мобільному додатку
        if username not in self.users:
            self.users[username] = {"device_id": device_id, "notifications": []}
            return f"Користувач {username} зареєстрований."
        return "Користувач уже існує."

    def send_notification(self, username, message):
# Надсилання push-сповіщення
        if username in self.users:
            self.users[username]["notifications"].append(message)
            self.notifications[username] = self.notifications.get(username, [])
+ [message]
            return f"Сповіщення відправлено {username}: {message}"
        return "Користувач не знайдений."

# Тестування MobileAPI
if __name__ == "__main__":
    api = MobileAPI()
    print(api.register_user("admin", "device_12345"))
    print(api.send_notification("admin", "Ваше з'єднання стабільне"))
```

Файл hardware_monitoring.py

```
import random
import time

# Клас для моніторингу апаратного стану точок доступу
class HardwareMonitor:

    def __init__(self, ap_name):
# Ініціалізація назви точки доступу
        self.ap_name = ap_name

    def get_temperature(self):
# Симуляція зчитування температури пристрою
        return random.uniform(30.0, 80.0)

    def get_cpu_load(self):
# Симуляція зчитування завантаження CPU
        return random.uniform(10.0, 95.0)

# Тестування HardwareMonitor
if __name__ == "__main__":
    monitor = HardwareMonitor("AP1")
    for _ in range(5):
        temp = monitor.get_temperature()
        cpu = monitor.get_cpu_load()
        status = "КРИТИЧНО!" if temp > 75 or cpu > 90 else "Нормально"
        print(f"{monitor.ap_name} - Температура: {temp:.2f}°C, CPU: {cpu:.2f}%,
Стан: {status}")
        time.sleep(2)
```

Файл `firmware_update.py`

```
import time

# Клас для керування оновленнями прошивки
class FirmwareUpdater:

    def __init__(self):
# Ініціалізація доступних версій прошивки
        self.current_version = "1.0.0"
        self.available_versions = ["1.0.1", "1.1.0", "2.0.0"]
        self.backup_version = None

    def check_for_updates(self):
# Перевірка доступних оновлень
        if self.available_versions:
            return f"Доступна нова версія: {self.available_versions[-1]}"
        return "Ви використовуєте останню версію."

    def backup_firmware(self):
# Створення резервної копії перед оновленням
        self.backup_version = self.current_version
        print(f"Резервна копія версії {self.backup_version} створена.")

    def update_firmware(self):
# Виконання оновлення
        if self.available_versions:
            self.backup_firmware()
            self.current_version = self.available_versions.pop()
            print(f"Оновлення завершено. Поточна версія:
{self.current_version}")
        else:
            print("Немає нових оновлень.")

# Тестування FirmwareUpdater
if __name__ == "__main__":
    updater = FirmwareUpdater()
    print(updater.check_for_updates())
    updater.update_firmware()
```

Файл `iot_support.py`

```
# Клас для керування підключенням IoT-пристроїв до Wi-Fi
class IoTManager:

    def __init__(self):
# Ініціалізація списку пристроїв
        self.devices = {}

    def register_device(self, device_id, device_type):
# Реєстрація нового IoT-пристрою
        if device_id not in self.devices:
            self.devices[device_id] = {"type": device_type, "firmware_version":
"1.0"}

            return f"Пристрій {device_id} ({device_type}) зареєстровано."
            return "Пристрій уже зареєстрований."

    def update_firmware(self, device_id):
# Оновлення ПЗ IoT-пристрою
        if device_id in self.devices:
            self.devices[device_id]["firmware_version"] = "1.1"
            return f"Прошивка пристрою {device_id} оновлена до 1.1"
            return "Пристрій не знайдено."

# Тестування IoTManager
if __name__ == "__main__":
    iot_manager = IoTManager()
    print(iot_manager.register_device("iot_001", "Smart Sensor"))
    print(iot_manager.update_firmware("iot_001"))
```