МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ АЕРОНАВІГАЦІЇ,
ЕЛЕКТРОНІКИ ТА ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ ТА РАДІОЕЛЕКТРОННИХ СИСТЕМ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри

_____ Віктор ГНАТЮК
“_____”_____2023 р.

КВАЛІФІКАЦІЙНА РОБОТА
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ МАГІСТР

Тема: «Алгоритмічне програмне забезпечення для передачі даних через бездротові децентралізовані мережі»

Виконавець: _____ Денис ПАВЛЕНКО
(підпис)

Керівник: _____ Віталій КУРУШКІН
(підпис)

Консультанти з окремих розділів пояснювальної записки:

Консультант розділу «Охорона праці» _____ Батир ХАЛМУРАДОВ
(підпис)

Консультант розділу «Охорона навколишнього середовища»
_____ Андріан ЯВНЮК
(підпис)

Нормоконтролер: _____ Денис БАХТІЯРОВ
(підпис)

Київ 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій
Кафедра телекомунікаційних та радіоелектронних систем
Спеціальність 172 «Телекомунікації та радіотехніка»
Освітньо-професійна програма «Телекомунікаційні системи та мережі»

ЗАТВЕРДЖУЮ
Завідувач кафедри

_____ Віктор ГНАТЮК
“_____”_____2023 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

Павленка Дениса Олександровича
(прізвище, ім'я, по батькові випускника в родовому відмінку)

1. Тема кваліфікаційної роботи: «Алгоритмічне програмне забезпечення для передачі даних через бездротові децентралізовані мережі»
затверджена наказом ректора від «28» вересня 2023 р. №1965/ст
2. Термін виконання роботи: з 02.10.2023 р. по 31.12.2023 р.
3. Вихідні дані до роботи: середовище для розробки додатків Android Studio, протоколи та алгоритми технології MANET
4. Зміст пояснювальної записки: аналітичний огляд доступних програмних рішень на ринку України та світу; проєктування алгоритмів маршрутизації; алгоритмічне програмне забезпечення для передачі даних через бездротові; децентралізовані мережі; тестування розробленого додатку на Android пристрої; охорона праці; охорона навколишнього середовища
5. Перелік обов'язкового графічного (ілюстративного) матеріалу: листинги коду алгоритмічного програмного забезпечення під ОС Android; зображення інтерфейсу алгоритмічного програмного забезпечення під ОС Android; слайди презентації MS Power Point на захист кваліфікаційної роботи.

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1	Розробити деталізований зміст розділів кваліфікаційної роботи	02.10.2023-04.10.2023	Виконано
2	Вступ	05.10.2023-08.10.2023	Виконано
3	Аналітичний огляд доступних програмних рішень на ринку України та світу	09.10.2023-22.10.2023	Виконано
4	Проєктування алгоритмів маршрутизації	23.10.2023-05.11.2023	Виконано
5	Алгоритмічне програмне забезпечення для передачі даних через бездротові децентралізовані мережі	06.11.2023-24.11.2023	Виконано
6	Тестування розробленого додатку на Android пристрої	25.11.2023-30.11.2023	Виконано
7	Охорона праці	01.12.2023-06.12.2023	Виконано
8	Охорона навколишнього середовища	07.12.2023-17.12.2023	Виконано
9	Усунення недоліків та захист кваліфікаційної роботи	18.12.2023-31.12.2023	Виконано

7. Консультанти з окремих розділів

Розділ	Консультант (посада, П.І.Б.)	Дата, підпис	
		Завдання видав	Завдання прийняв
Охорона праці	к.м.н., професор Батир ХАЛМУРАДОВ		
Охорона навколиш-нього середовища	к.б.н., доц. Андріан ЯВНЮК		

8. Дата видачі завдання: “29" вересня 2023 р.

Керівник кваліфікаційної роботи _____ Віталій КУРУШКІН
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання _____ Денис ПАВЛЕНКО
(підпис випускника) (П.І.Б.)

РЕФЕРАТ

Кваліфікаційна робота «Алгоритмічне програмне забезпечення для передачі даних через бездротові децентралізовані мережі» містить 84 сторінки, 28 рисунків, 14 таблиць, 52 використаних джерела.

БЕЗДРОТОВІ МЕРЕЖІ, ДЕЦЕНТРАЛІЗОВАНА МЕРЕЖА, АЛГОРИТМИ ПЕРЕДАЧІ ДАНИХ, ПРОТОКОЛИ МАРШРУТИЗАЦІЇ, МЕРЕЖЕВИЙ СТЕК, МЕРЕЖЕВИЙ АНАЛІЗ, ДИНАМІЧНЕ ПЕРЕРОЗПОДІЛЕННЯ РЕСУРСІВ, ВУЗЛИ МЕРЕЖІ, МЕРЕЖЕВА БЕЗПЕКА, ОПТИМІЗАЦІЯ ПРОПУСКНОЇ ЗДАТНОСТІ, ДЕКІЛЬКА ШАРІВ OSI, АНТЕННІ ТЕХНОЛОГІЇ, ІНТЕРНЕТ РЕЧЕЙ (IOT), ПЕРЕДАЧА ВЕЛИКОГО ОБСЯГУ ДАНИХ, ТЕХНОЛОГІЇ МЕРЕЖЕВОГО ВІДДАЛЕНОГО КЕРУВАННЯ, ЗЕЛЕНА МЕРЕЖА, ТОПОЛОГІЯ МЕРЕЖІ, ІНТЕРФЕЙС КОРИСТУВАЧА ДЛЯ НАЛАШТУВАННЯ МЕРЕЖІ, МЕРЕЖЕВІ СТАНДАРТИ, QOS (ЯКІСТЬ ОБСЛУГОВУВАННЯ) В БЕЗДРОТОВИХ МЕРЕЖАХ.

Метою кваліфікаційної роботи є розробка алгоритмічного програмного забезпечення під операційну систему Android для передавання даних, використовуючи технологію Wi-Fi Direct, спираючись на протоколи та алгоритми технології MANET.

Об'єктом дослідження є самі бездротові децентралізовані мережі та процес передачі даних в таких мережах.

Предметом дослідження є конкретні алгоритми та програмне забезпечення, які використовуються для оптимізації та забезпечення передачі даних через бездротові децентралізовані мережі.

Результати дослідження мають практичне застосування в різних галузях і можуть сприяти покращенню ефективності та надійності передачі даних через бездротові децентралізовані мережі в різних сферах життя.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ

WLAN - Wireless Local Area Network (Бездротова локальна мережа).

MANET - Mobile Ad Hoc Network (Мобільна бездротова самоорганізована мережа).

IoT - Internet of Things (Інтернет речей).

MAC - Media Access Control (Керування доступом до медіа).

IP - Internet Protocol (Протокол Інтернету).

TCP - Transmission Control Protocol (Протокол керування передачею).

UDP - User Datagram Protocol (Протокол передачі користувацьких дейтаграм).

OSI - Open Systems Interconnection (Модель відкритого системного взаємодії).

QoS - Quality of Service (Якість обслуговування).

SSID - Service Set Identifier (Ідентифікатор набору служби).

LAN - Local Area Network (Локальна мережа).

WAN - Wide Area Network (Широкомасштабна мережа).

RF - Radio Frequency (Радіочастота).

NLOS - Non-Line-of-Sight (Непрямий зв'язок).

LOS - Line-of-Sight (Прямий зв'язок).

MIMO - Multiple Input Multiple Output (Багатоантенний вхід-вихід).

SDR - Software-Defined Radio (Програмно-визначений радіо).

DSR - Dynamic Source Routing (Динамічна маршрутизація від вихідного джерела).

GPS - Global Positioning System (Глобальна система позиціонування).

AP - Access Point (Точка доступу).

IEEE - Institute of Electrical and Electronics Engineers (Інститут інженерів електротехніки та електроніки).

CSMA/CA - Carrier Sense Multiple Access with Collision Avoidance (Засіб багатократного доступу з перевіркою несучої та уникненням зіткнень).

PHY - Physical Layer (Фізичний рівень).

OSI - Open Systems Interconnection (Модель відкритого системного взаємодії).

IoT - Internet of Things (Інтернет речей).

WPA - Wi-Fi Protected Access (Захищений доступ Wi-Fi).

VPN - Virtual Private Network (Віртуальна приватна мережа).

SDN - Software-Defined Networking (Програмно-визначена мережева структура).

NAT - Network Address Translation (Переклад адрес мережі).

TTL - Time to Live (Час життя).

ICMP - Internet Control Message Protocol (Протокол керування повідомленнями в мережі Інтернет).

SS7 - Signaling System No. 7 (Сигнальна система № 7).

LOS - Level of Service (Рівень обслуговування).

APN - Access Point Name (Ім'я точки доступу).

AES - Advanced Encryption Standard (Стандарт передового шифрування).

CDMA - Code Division Multiple Access (Багатозонна дільниця за кодом).

IoT - Internet of Things (Інтернет речей).

DNS - Domain Name System (Система доменних імен).

JSON - JavaScript Object Notation (Формат обміну даними).

DHCP - Dynamic Host Configuration Protocol (Протокол динамічної конфігурації хоста).

ВСТУП

Нині інформатизація суспільства досягла високих результатів і цьому більшою мірою посприяли мобільні технології. Однією з найважливіших функцій сучасного гаджета є передача або отримання графічної та текстової інформації, а також файлів мультимедії. Ця функція робить мобільні пристрої залежними від інтернету, і у зв'язку з цим постає питання про передачу даних без наявності точок доступу підключення до глобальної мережі.

Актуальність теми «Алгоритмічне програмне забезпечення для передачі даних через бездротові децентралізовані мережі» в сучасному світі важко переоцінити, оскільки вона відображає ключовий аспект розвитку та використання бездротових комунікаційних технологій, що обґрунтовується наступними положеннями:

- *Зростання використання бездротових технологій*. З кожним роком використання бездротових технологій продовжує зростати, включаючи бездротовий доступ до Інтернету, IoT-пристрої, мобільні додатки та бездротові мережі. Це створює підвищену потребу в ефективних алгоритмах передачі даних через такі мережі.
- *Споживачі вимагають більше пропускної здатності та якості обслуговування.* Споживачі сподіваються на високу пропускну здатність та безперебійну роботу у бездротових мережах, особливо в місцях густої населеності. Це ставить завдання перед розробниками алгоритмів оптимізації мереж та передачі даних.
- *Мережі Інтернету речей (IoT)*. Інтернет речей набуває обертів у багатьох галузях, включаючи медицину, транспорт, промисловість та побутову сферу. Ефективна передача даних через бездротові децентралізовані мережі є критично важливою для цих застосувань.
- *Розвиток безпеки в мережах*. З підвищенням кількості безпекових загроз і кібератак, алгоритми для бездротових децентралізованих мереж також мають бути розроблені з урахуванням підвищених вимог до безпеки.

- *Мобільний інтернет та 5G*. Впровадження технології 5G та мобільний інтернет робить передачу даних ще швидшою і більш ефективною. Це вимагає нових алгоритмів для оптимізації мереж та забезпечення надійної передачі даних.
- *Екологічна дружність*. Розвиток алгоритмів, які дозволяють ефективну передачу даних в бездротових мережах, сприяє зменшенню споживання енергії та забезпечує більшу екологічну дружність мережових інфраструктур.

Однією з відносно нових технологій для передачі даних без маршрутизаторів є Wi-Fi Direct.

Wi-Fi Direct - стандарт (набір програмних протоколів), що дають змогу двом і більше Wi-Fi пристроям спілкуватися один з одним без маршрутизаторів і хот-спотів [1-38].

За допомогою мобільного застосунку та наявності в мобільному пристрої технології Wi-Fi Direct, яка наразі присутня практично на кожному смартфоні, з'являється можливість передавати дані каналами зв'язку, які створюються між користувачами самого застосунку.

Метою кваліфікаційної роботи є розробка алгоритмічного програмного забезпечення під операційну систему Android для передавання даних, використовуючи технологію Wi-Fi Direct, спираючись на протоколи та алгоритми технології MANET.

Для досягнення мети кваліфікаційної роботи було поставлено **такі завдання**:

- Знайти та проаналізувати наявні аналоги.
- Проаналізувати наявні протоколи маршрутизації в мережах MANET.
- Спроєктувати алгоритми для передачі даних за допомогою технології Wi-Fi Direct.
- Програмно реалізувати спроєктовані алгоритми.
- Протестувати отримане програмне забезпечення.

Об'єктом дослідження є самі бездротові децентралізовані мережі та процес передачі даних в таких мережах.

Предметом дослідження є конкретні алгоритми та програмне забезпечення, які використовуються для оптимізації та забезпечення передачі даних через бездротові децентралізовані мережі.

Для проведення дослідження на тему «Алгоритмічне програмне забезпечення для передачі даних через бездротові децентралізовані мережі» використовувались різні **методи дослідження**. Ось деякі з них:

- *Аналіз літератури.* Перший етап дослідження може включати вивчення наукових статей, книг, журнальних публікацій та інших джерел, що стосуються теми. Це дозволить отримати обсяг існуючих даних та розуміння сучасного стану галузі.
- *Емпіричні дослідження.* Можливо проведення практичних досліджень та експериментів для збору даних та отримання результатів в реальних умовах. Це може включати створення демонстраційних систем, тестування алгоритмів на симуляційних платформах або проведення полігонних випробувань.
- *Математичне моделювання.* Математичне моделювання може бути використано для розрахунку характеристик бездротових мереж та алгоритмів передачі даних. Це допоможе визначити ефективність та пропускну здатність мережі в залежності від різних параметрів.
- *Аналіз даних.* Для обробки та аналізу даних можна використовувати статистичні методи, машинне навчання та інші інструменти. Це допоможе отримати інсайти зі зібраних даних та оцінити ефективність алгоритмів.
- *Порівняльний аналіз.* Можливо проведення порівняльного аналізу різних алгоритмів та підходів для передачі даних в бездротових мережах, щоб визначити їх переваги та недоліки.
- *Симуляція та моделювання.* Використання спеціалізованих програмних засобів для створення симуляцій моделей бездротових мереж, які дозволяють віртуально тестувати алгоритми та програмне забезпечення в різних умовах.

- *Аналіз відкритих даних та інформаційних ресурсів*. Використання відкритих даних та інформаційних ресурсів, таких як документація від виробників обладнання та програмного забезпечення, може стати важливим джерелом інформації для дослідження.

Наукова новизна отриманих результатів:

- Набуло подальшого розвитку використання існуючих алгоритмів для підвищення пропускної здатності мережі та зменшення перешкод у передачі даних.

Практичне значення отриманих результатів:

- *Телекомунікації.* Результати дослідження можуть бути використані для покращення передачі даних у бездротових мережах операторів мобільного зв'язку, зокрема для збільшення пропускної здатності та надійності мереж.
- *Інтернет речей (IoT).* Розроблені алгоритми та програмне забезпечення можуть бути використані для підтримки передачі даних в мережах IoT, що стає все більш важливим у сферах моніторингу, автоматизації та розумних системах.
- *Медичні додатки.* Вдосконалені алгоритми та програмне забезпечення можуть сприяти покращенню збору та передачі медичних даних в бездротових мережах, що є важливим для телемедицини та моніторингу пацієнтів.
- *Автономні автомобілі та транспорт.* Результати дослідження можуть бути застосовані в розвитку систем комунікації та передачі даних для автономних автомобілів та інших транспортних засобів.

Апробація отриманих результатів. Основні положення роботи доповідалися та обговорювалися на таких конференціях:

- Науково-практична конференція «Проблеми експлуатації та захисту інформаційно-комунікаційних систем», м. Київ, 2023 р.

РОЗДІЛ 1
АНАЛІТИЧНИЙ ОГЛЯД ДОСТУПНИХ ПРОГРАМНИХ РІШЕНЬ НА РИНКУ УКРАЇНИ ТА СВІТУ

1.1. Аналіз аналогів програмного продукту

На даний момент на ринку мобільних додатків не так багато програмних продуктів, які працюють з технологією Wi-Fi Direct, і, в основному, реалізують лише користувальницький інтерфейс для передачі даних для двох мобільних пристроїв. Існують додатки, які розширюють можливості користувача і пропонують додаткову інформацію про Wi-Fi сигнал, даючи змогу вибрати точку з найкращою пропускною спроможністю. З основних аналогів програмних продуктів, які працюють під операційною системою Android і є продуктами сервісу "Google Play Market" хотілося б виділити: Swift-Wi-Fi і SuperBeam [23].

Swift Wi-Fi. Swift Wi-Fi - додаток розроблений групою розробників "Swift Wi-Fi Team". У цьому додатку крім підключення пристроїв за допомогою Wi-Fi Direct не малу роль відіграє апаратна частина у вигляді додаткових точок доступу, контролерів і антен. Так само містить у собі віджет, який показує, як доступні мережі WiFi, так і дані про них [24].

Віджет дає змогу виконати такі функції:

1. Перемикання Wi-Fi за допомогою кнопки ВКЛ/ВИКЛ;
2. Підключення до потрібної Wi-Fi точки доступу;
3. Під час натискання на точку доступу, показує безпеку точки доступу, швидкість з'єднання, силу сигналу та IP-адресу;
4. Керування Wi-Fi точкою доступу (Видалити невикористовувані точки доступу);
5. Додати нове Wi-Fi з'єднання (підтримує Open, WEP, WPA/WPA2 PSK і 802.1x EAP);
6. Легке налаштування за допомогою QRCode (Права Root обов'язкові).

SuperBeam Wi-Fi Direct. SuperBeam - додаток, розроблений групою розробників "LiveQoS". дозволяє просто передавати дані між пристроями Android, використовуючи технологію Wi-Fi Direct. Дає можливість зв'язати між собою два пристрої, використовуючи QR-коди (за допомогою наявного в програмі сканера QR-кодів) або NFC, якщо пристрої її підтримують, або передаючи ключ вручну [25].

Основними функціями програми SuperBeam є:

- Зв'язування пристроїв з використанням NFC або сканування QR-коду;
- Передача даних через web-інтерфейс на пристрої, на яких не встановлено SuperBeam;
- Передача одного або декількох файлів будь-якого типу (фотографії, відео, zip-файли, APK, контакти тощо);
- Історія всіх виконаних операцій передачі;
- Автоматичне перемикання в режим точки доступу під час роботи з пристроями, які не підтримують WiFi direct;
- У разі збоїв у роботі WiFi direct може бути використано вже наявне мережеве WiFi-з'єднання між пристроями;
- Для більшої сумісності з обмеженими WiFi-мережами для всіх видів передачі використовується порт 8080;
- Працює з усіма сторонніми файловими менеджерами і фотогалереями.

З вищесказаного можна зробити висновок, що додатки, які використовують WiFi-Direct як основну функцію, передають дані лише між двома пристроями і нічим один від одного не відрізняються, за винятком різного інтерфейсу. Створюваний застосунок буде заснований на алгоритмах мереж Manet, з подальшою можливістю модифікації до передачі даних між двома і більше пристроями.

1.2. Аналіз протоколів маршрутизації

Wi-Fi Direct відносно нова технологія для передавання даних, але в неї так само є і конкурент - технологія MANET [1].

MANET (Mobile Ad hoc NETworks) мережі - радіомережі з випадковими мобільними абонентами, що реалізують повністю децентралізоване управління за відсутності базових станцій або опорних вузлів. Топологія - швидко мінлива з випадковим з'єднанням вузлів.

Сьогодні переважна більшість наземних мобільних бездротових мобільних мереж зв'язку мають фіксовану інфраструктуру та з'єднані між собою за допомогою різних, як правило дротових або радіорелейних, каналів передачі даних. В останнє десятиліття велика увага приділяється створенню мобільних пакетних радіомереж, які не мають фіксованої інфраструктури, - мережі стаціонарних (Ad Hoc) і мобільних абонентів (MANET) [2].

Такі мережі є самоорганізаційними, оскільки їхні вузли є не тільки кінцевими користувацькими терміналами, а й є ретрансляторами-маршрутизаторами, ретранслюючи пакети інших абонентів і беручи участь у знаходженні маршрутів до них, отже, ці мережі здатні до самоорганізації. Такі мережі можуть складатися з десятків, сотень і навіть тисяч вузлів.

На противагу мережам з ієрархічною структурою і централізованим управлінням, однорангові мережі без інфраструктури складаються з однотипних вузлів, де кожен вузол має комплекс програмно-апаратних засобів. Це дає змогу організувати передачу даних від джерела до одержувача безпосередньо за фізичної наявності такого шляху і тим самим розподілити навантаження на мережу та підвищити сумарну пропускну здатність мережі. Передача даних від одного абонента до іншого може відбуватися, навіть якщо ці вузли перебувають поза зоною прямої радіовидимості. У цих випадках пакети даних цих абонентів ретранслюються іншими вузлами мережі, які мають зв'язок із абонентами, що кореспондують. Мережі з багаторазовою ретрансляцією називаються багатопролітними або багатоскачковими (multihop). Під час розроблення таких мереж основними проблемами є маршрутизація пакетів від вузла джерела до вузла одержувача, масштабованість мереж, адресація кінцевих пристроїв, підтримання зв'язності в умовах змінної топології [3].

Технологія MANET базується на великій кількості протоколів маршрутизації зі складними алгоритмами для будь-яких ситуацій. До таких протоколів належать: DSR,

AODV, OLSR. Кожен із цих протоколів має свій алгоритм пошуку маршруту, передавання даних, а також власну структуру даних [4].

Незважаючи на те, що технологія MANET ще не використовується, і всі протоколи перебувають в експериментальному стані, їх можна частково використати й адаптувати під розроблювану систему.

Протокол DSR. Протокол DSR (Dynamic Source Routing Protocol) є простим і ефективним протоколом маршрутизації та призначений для роботи з бездротовими пристроями.

Протокол DSR регулярно оновлює свій кеш маршрутів, якщо в мережі з'явилися нові доступні прості маршрути. У разі, якщо було знайдено нові доступні маршрути, вузол надсилає пакет на цей маршрут. Пакет має знати про напрямок маршруту. Для цього інформація про маршрут вказується в пакеті, щоб досягти адресата від його джерела. DSR має дві основні функції для правильної роботи: запит маршруту та обслуговування маршруту. Під час відкриття маршруту, він має два повідомлення, запит маршруту (RREQ) і відповідь на запит маршруту (RREP). Коли вузол хоче надіслати повідомлення в певний пункт призначення, він транслює RREQ пакет у мережі. Сусідні вузли в діапазоні мовлення отримують це RREQ-повідомлення і додають свою власну адресу, і знову ретранслюють його в мережі. У разі якщо RREQ-повідомлення не доходить до вузла-адресата, маршрут видаляється [5].

Кожен вузол підтримує свій кеш маршруту, який зберігається в пам'яті для виявленого маршруту. Вузол перевірятиме свій кеш маршруту для необхідного вузла призначення до ретрансляції RREQ-повідомлення. Підтримуючи кеш маршруту на кожному вузлі в мережі, знижуються накладні витрати пам'яті, які генеруються за допомогою процедури виявлення маршруту. Якщо маршрут перебуває в цьому кеші маршрутів вузла, то він не буде ретранслювати RREQ у всій мережі. RREQ-повідомлення буде відразу транслюватися до вузла призначення. Перше повідомлення, що досягло адресата, має повну інформацію про маршрут. Цей вузол надсилатиме пакет RREP до джерела, що має повну інформацію про маршрут. Цей маршрут вважатиметься найкоротшим шляхом, яким пакет RREQ транслювався. Вузол-джерело тепер

має повну інформацію про маршрут у своєму кеші маршрутів і може починати передачу пакетів [6].

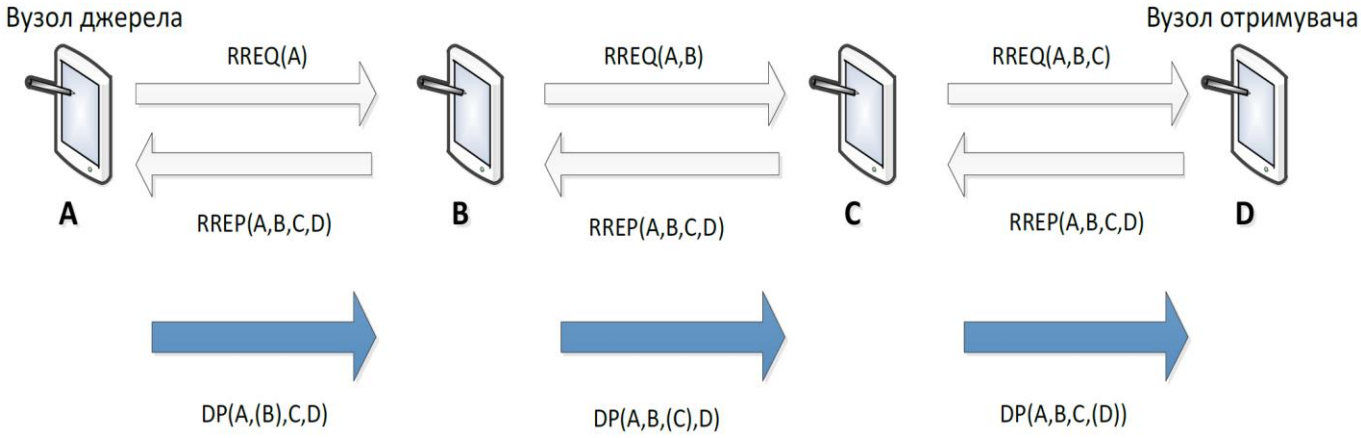

Рис. 1.1. Пошук маршруту за протоколом DSR

На Рис. 1.1. показано процедуру пошуку маршруту. Тут ми маємо чотири вузли, A, B, C і D, вузол A є джерелом, вузол D є призначенням. Коли вузол A бажає послати пакет даних у вузол D, він спочатку перевірить свій кеш маршруту, чи має він прямий маршрут до вузла D чи ні. Якщо вузол A не має прямого маршруту до вузла D, то він транслюватиме повідомлення RREQ у мережі. Сусідній вузол B отримає повідомлення RREQ. Вузол B перевіряє свій кеш маршруту, чи є прямий маршрут до вузла призначення D чи ні, якщо він знаходить маршрут до вузла призначення D, він посилає RREP повідомлення до вузла A.

У відповідь на це повідомлення вузол джерело почне посилати пакети даних (DP) за виявленим маршрутом. Якщо він не виявив маршрут від вузла B до вузла D, він передає повідомлення RREQ до наступного вузла C і зберігає маршрут A,B у кеші. Процес триває доти, доки повідомлення RREQ не досягне цільового вузла D. Вузол призначення D кешує маршрути AB, BC і CD у своїй пам'яті і посилає повідомлення RREP до вихідного вузла A.

Наступною функцією є обслуговування маршруту. Обслуговування маршруту використовує два види повідомлень: повідомлення про помилку (RERR) і підтвердження (ACK). ACK відправляє повідомлення вузлу-джерелу, якщо вузол призначення було досягнуто. Усі проміжні вузли так само отримують інформацію про новий

побудований маршрут. Якщо в мережі сталася помилка в процесі пошуку маршруту, вузлу-джерелу надходить повідомлення про помилку (RRER). Це відбувається, якщо вузол-джерело не зміг отримати АСК-повідомлення.

Таким чином, джерело отримує пакет RERR для того, щоб знову надіслати запит на пошук маршруту. Отримавши повідомлення RERR, вузли видалити записи маршруту [7].

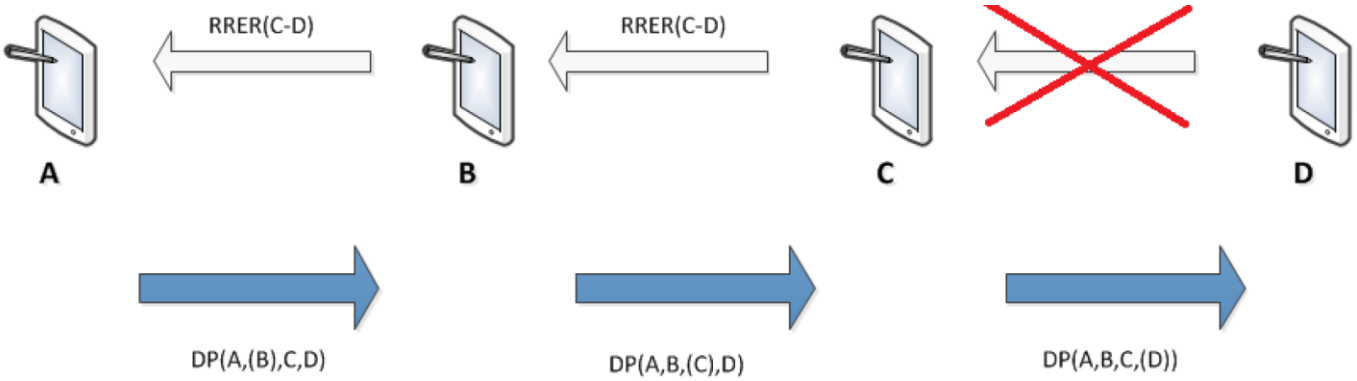

Рис. 1.2. Повідомлення про підтвердження та помилку в протоколі DSR

На Рис. 1.2. показано чотири вузли, тобто A, B, C і D. Вузол А надсилає повідомлення до вузла призначення D. Повідомлення проходить до вузла С. Коли вузол С пересилає повідомлення RREQ стосовно вузла D, і не отримує повідомлення про підтвердження АСК від вузла Д, вузол С розуміє, що з'явилися проблеми під час передачі.

Унаслідок чого, вузол С посилає повідомлення RRER до вихідного вузла А, що дає змогу вузлу А почати новий пошук маршруту до вузла D [8].

Протокол AODV. AODV (Ad hoc On-Demand Distance Vector) є протоколом маршрутизації на вимогу. Алгоритм AODV дає простий спосіб отримати інформацію про зміну ситуації під час відправлення пакетів даних. Таким чином, якщо відправлення пакета не вдалося, виявляється уражений вузол і всім постраждалим від цього вузлам надсилається повідомлення. Це повідомлення скасовує всі маршрути через уражений вузол. В AODV маршрути будуються односпрямовані, від джерела до місця призначення, і тому використання мережі є мінімальним. Оскільки маршрути будуються на вимогу, мережевий трафік так само буде мінімальним. AODV не дозволяє

зберігати додаткові маршрути, які не використовуються. Якщо два вузли хочуть встановити з'єднання в одноранговій мережі, то AODV дозволяє їм побудувати багатоінтервальний маршрут. AODV використовує порядковий номер вузла-одержувача (DSN), щоб уникнути підрахунку до нескінченності [9].

Коли вузол-джерело відправляє запит до пункту призначення, він надсилає свої DSNs разом з усією інформацією про маршрутизацію. Він також вибирає найвигідніший маршрут, ґрунтуючись на кількості DSN.

Існує три AODV-повідомлення: запит маршруту (RREQs), відповідь на запит маршруту (RREPs), і помилка маршруту (RERRs).

Коли вузол-джерело хоче створити новий маршрут до місця призначення, вузол-запитувач транслює RREQ-повідомлення найближчим сусідам.

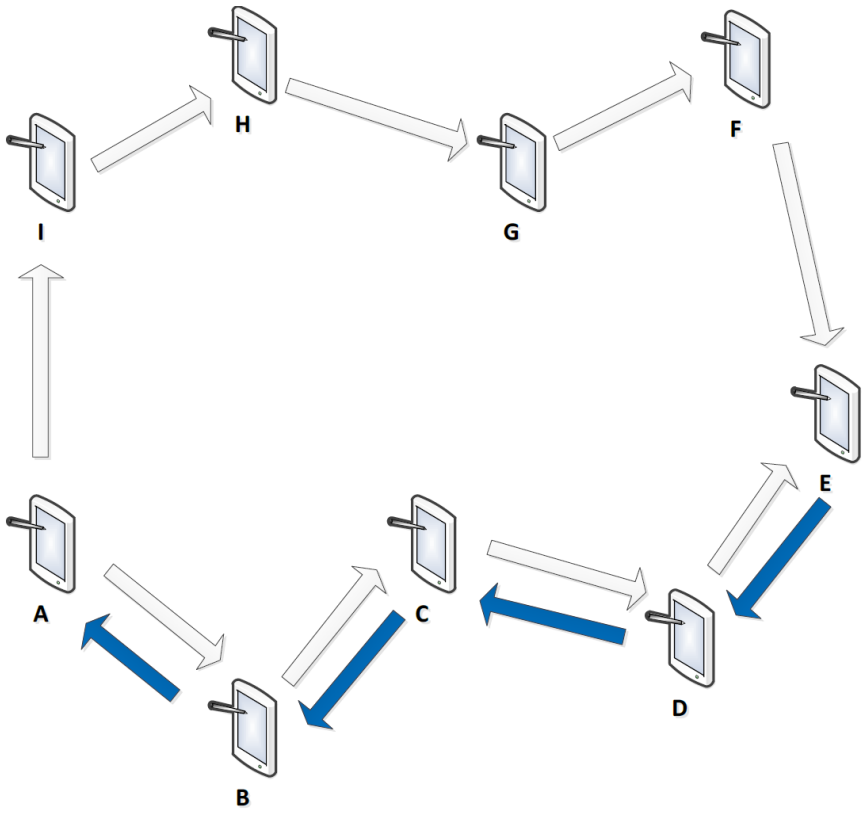

Рис. 1.3. Пошук маршруту в протоколі AODV

На Рис. 1.3 повідомлення RREQ транслюється з вузла-джерела A до вузла призначення E. Повідомлення RREQ показано білою стрілкою від вузла-джерела A за двома напрямками. Вихідний вузол A транслює повідомлення RREQ сусіднім вузлам.

Кожен доданий вузол на шляху до вузла призначення збільшує значення DSN на одиницю. Кожен проміжний вузол буде перевіряти чи має він прямий маршрут до місця призначення чи ні. Якщо у нього є маршрут, він направить RREP до джерела вузла A. Якщо він не має прямого маршруту до місця призначення, він транслюватиме повідомлення RREQ у мережі знову, зі зміненим значенням порядкового номера DSN.

На малюнку 3 показано процедуру для знаходження вузла призначення E. повідомлення RREQ транслюється в мережі для знаходження вузла призначення. Проміжні вузли можуть відповісти на повідомлення RREQ, тільки якщо вони мають число DSN, що дорівнює або більше, ніж кількість вмісту в заголовку пакета з RREQ. Проміжні вузли пересилають повідомлення RREQ до сусідніх вузлів і записують адреси цих вузлів у кеші маршрутизації. Цю інформацію використовуватимуть, щоб зробити зворотний шлях для RREP повідомлення від вузла призначення, як показано на рис. 1.3.

Вузол призначення E відповідає повідомленням RREP, позначеним синіми стрілками, який позначає найкоротший шлях від вузла призначення E до вузла-джерела A. Вузли, які отримують RREP повідомлення кешують інформацію про побудований маршрут.

Якщо пошук маршруту не дав результату, генерується повідомлення RERR. RERR повідомлення містить інформацію про вузли, які не досяжні.

Уся інформація про маршрутизацію в мережі зберігається в таблиці. Таблиця маршрутизації надана в таблиці 1: IP-адреса призначення, порядковий номер (DSN), послідовність вузлів до вузла призначення, мережевий інтерфейс, кількість переходів, тайм аут [10].

Таблиця 1.1

Таблиця маршрутизації

IP-адреса призначення	DSN	Послідовність вузлів до вузла призначення	Мережевий інтерфейс	Кількість переходів	Тайм-аут

Протокол OLSR. OLSR (Optimized Link-State Routing) це проактивний протокол маршрутизації. OLSR так само називають табличним протоколом, оскільки він постійно зберігає і оновлює свою таблицю маршрутизації [11].

OLSR відстежує таблиці маршрутизації, для того щоб забезпечити оптимальний маршрут, під час надсилання запиту на передачу даних до вузла призначення. Завдяки своїй природі OLSR називається проактивним протоколом маршрутизації. Multipoint Relay (MPR) вузли показані на рис. 1.4 [12].

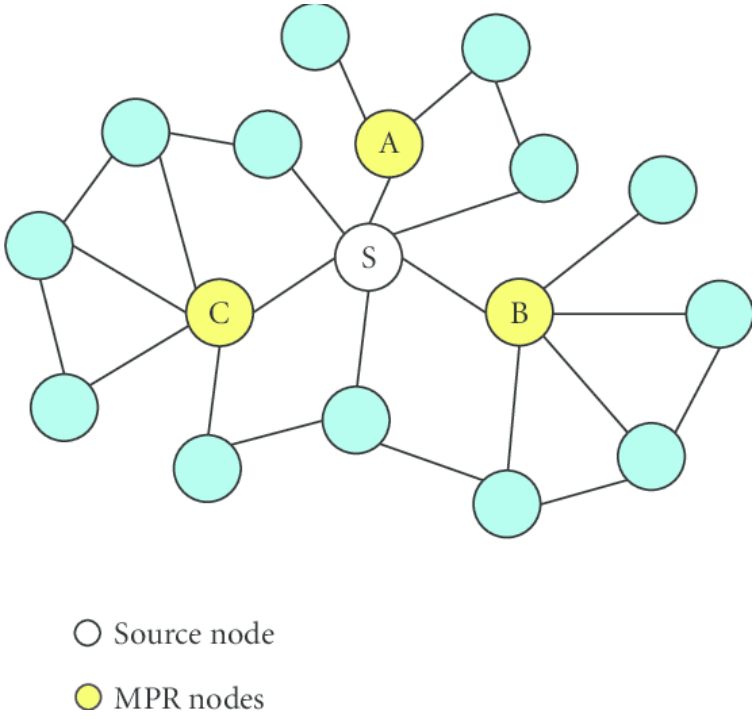

Рис. 1.4. Multipoint Relay в OLSR

В OLSR не всі вузли обмінюються інформацією один з одним. Multipoint Relay (MPR) вузли надають широкомовну передачу пакетів з маршрутами. Вузол MPR вибирають як сусіда для вузла-джерела. Кожен вузол у мережі зберігає список вузлів MPR [13].

Селектор MPR отримують із надсилання пакетів "Hello" між сусідніми вузлами. Ці маршрути побудовані до того, як вихідний вузол має намір направити повідомлення за вказаним призначенням. Кожен вузол у мережі зберігає таблицю маршрутизації. Саме з цієї причини маршрутизації накладні витрати на OLSR мінімальні порівняно з іншими протоколами реактивної маршрутизації і забезпечують найкоротший

маршрут до місця призначення в мережі. Немає необхідності будувати нові маршрути, оскільки наявні маршрути мають запасні. Це зменшує затримку виявлення нового маршруту. Вузли в мережі відправляють повідомлення "Hello" своїм сусідам. Ці повідомлення надсилаються із заданим інтервалом в OLSR для визначення стану каналу зв'язку [14].

Якщо вузол А і вузол В є сусідами, вузол А надсилає повідомлення "Hello" у вузол В. Якщо вузол В приймає це повідомлення, ми можемо сказати, що зв'язок є асиметричним. Якщо вузол В надсилає те саме повідомлення "Hello" до вузла, то це те ж саме, як у першому випадку, що називається асиметричним запитом. Виходячи з цього, встановлюється двосторонній зв'язок і ми можемо назвати цей зв'язок симетричним. Повідомлення "Hello" містить усю інформацію про сусідів. Це дозволяє будь-якому вузлу мати таблицю, в якій він має інформацію про всіх своїх численних сусідів. Вузол вибирає мінімальну кількість вузлів MPR, коли симетричний зв'язок встановлено [15].

1.3. Порівняльний аналіз протоколів маршрутизації

Порівняльний аналіз основних характеристик реактивних протоколів подано в таблиці 1.2.

Таблиця 1.2

Порівняння реактивних протоколів

Ім'я	Маршрутна структура	Множинні маршрути	Метод розрахунку маршруту	Зберігання маршрутів	Стратегія
AODV	Плоска	Ні	Новий найкоротший шлях	Маршрутна таблиця	Повний або частковий перезапис маршрутів
DSR	Плоска	Так	Найкоротший шлях або наступний у маршрутній таблиця	Кеш маршрутів	Повний перезапис маршрутів

Порівняльний аналіз складності протоколів маршрутизації подано в таблиці 1.3.

Таблиця 1.3

Складність проактивних і реактивних протоколів

Ім'я	Час конвергенції	Пам'ять	Управління	Переваги \ недоліки
OLSR	O(D*I)	$O(N)^2$	$O(N)^2$	Захист від петель на підставі попередньої інформації / необхідність обміну даними
AODV	O(2D)	O(2N)	O(2N)	Адаптується до високодинамічних топологій
DSR	O(2D)	O(2N)	O(2N)	Безліч маршрутів, неможливо прослухати

де O() - порядок складності; D - діаметр мережі; I - середній час оновлення; N - кількість вузлів у мережі [16-19].

Порівняння переваг і недоліків протоколів різного типу представлено в таблиці 1.4.

Таблиця 1.4

Переваги та недоліки протоколів OLSR і AODV

OLSR	**AODV**
Проактивний протокол	Реактивний протокол
зберігає новітні дані про всі мережі	зберігає дані тільки про активні шляхи, завдяки чому менше оброблення службової інформації
час знаходження нового шляху менший, оскільки необхідно тільки переглянути маршрутну таблицю, яка постійно оновлюється	час знаходження нового шляху більший, оскільки інформація є тільки про активні маршрути
більш ефективний для мереж із меншою мобільністю	більш ефективний для мереж із високою мобільністю

У результаті порівняння, найбільш оптимальним протоколом є DSR. Він включає в себе просте зберігання великої кількості маршрутів, може бути застосований для мереж з високою мобільністю. DSR може бути використаний як основа для розробки алгоритмів для передачі даних за допомогою технології Wi-Fi Direct [20-22].

РОЗДІЛ 2
ПРОЄКТУВАННЯ АЛГОРИТМІВ МАРШРУТИЗАЦІЇ

2.1. Алгоритм пошуку маршруту

Алгоритм пошуку маршруту дає змогу зібрати необхідну інформацію для побудови маршруту між джерелом і одержувачем.

Маршрут може бути знайдений двома способами:

- може бути знайдений у кеші маршрутизації, якщо цей маршрут будують не вперше і дані маршруту не встигли застаріти;
- може бути знайдений за допомогою запиту на пошук маршруту, надісланого найближчим сусідам, який ретранслюватиметься від сусіда до сусіда доти, доки маршрут не буде знайдений або доки не закінчиться час життя запиту.

Якщо маршрут міститься в кеші маршрутизації проміжного вузла, то запит на пошук маршруту не буде ретрансльовано по всій мережі. Перше повідомлення, що досягло адресата, має повну інформацію про маршрут. Автоматично цей маршрут вважатиметься найкоротшим і найвигіднішим для передачі даних.

Якщо маршруту немає в кеші маршрутизації, необхідно його побудувати за допомогою розсилки запиту на пошук маршруту всім найближчим сусідам.

Якщо сусідні вузли не можуть знайти у своїй зоні видимості вузол-одержувач, то кожен вузол ретранслює запит своїм найближчим сусідам, і так триває доти, доки маршрут не буде знайдений, або доки не спливе час життя пакета.

Кожен вузол окремо обробляє запит і повертає вузлу-джерелу або інформацію про шлях, або про те, що маршрут побудувати не вдалося. У зв'язку з цим з'являється можливість знаходження декількох маршрутів через різні вузли, що дасть змогу записати в кеш маршрутизації кілька альтернативних вузлів. Найбільш пріоритетним буде маршрут, у якому найменша кількість проміжних вузлів.

Блок-схему алгоритму пошуку маршруту надано на рис. 2.1.

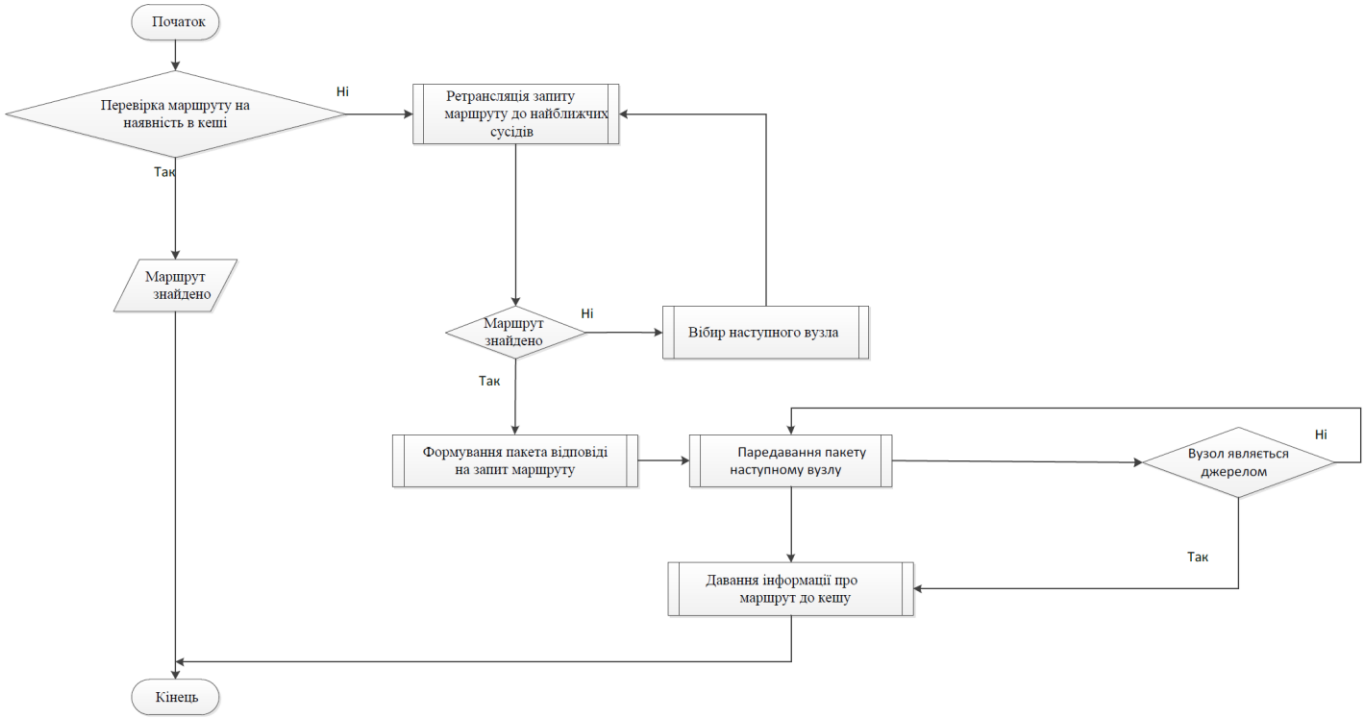

Рис. 2.1. Блок-схема алгоритму пошуку маршруту

Інформація, яку несе пакет запиту маршруту, показана в таблиці 2.1 [26].

Таблиця 2.1

Структура даних пакета запиту маршруту

1 байт	16 байт	16 байт	16 байт	8 байт	8 байт	16*N байт
Тип запиту	UID	Вузол-джерело	Вузол - одержувач	Тайм-аут	Hops	MAC-адреси проміжних вузлів

Поле "Тип запиту" містить бінарне число, що відповідає типу пакета. Для пакета запиту маршруту за замовчуванням 0001;

Поле "UID" містить унікальний ідентифікатор створеного запиту;

Поле "Вузол-джерело" містить інформацію про вузол, який ініціював запит пошуку маршруту;

Поле "Вузол-одержувач" містить інформацію про вузол, маршрут до якого необхідно побудувати;

Поле "Тайм-аут" містить час життя пакета. Під час створення запиту

"Тайм-аут" за замовчуванням дорівнює 30 секундам. При кожному збільшенні поля "Hops", час відповідно збільшується на 30 секунд. Якщо час минув, пакет видаляється;

Поле "Hops" містить бінарне число, за замовчуванням 0000. При кожній передачі пакета на інший вузол, значення збільшується на 0001;

У полі "MAC-адреси проміжних вузлів" кожен проміжний вузол додає свою MAC-адресу.

За досягнення вузла-одержувача, формується відповідь на запит маршруту, використовуючи кінцевий варіант пакета запиту маршруту.

Інформація, яку несе пакет відповіді на запит маршруту, представлена в таблиці 2.2 [27].

Таблиця 2.2

Структура даних пакета відповіді на запит маршруту

1 байт	16 байт	16 байт	16 байт	8 байт	8 байт	16*N байт
Тип запиту	UID	Вузол-джерело	Вузол-одержувач	Тайм-аут	Hops	MAC-адреси проміжних вузлів

Поле "Тип запиту" містить бінарне число, що відповідає типу пакета. Для пакета відповіді на запит маршруту за замовчуванням 0010;

Поле "UID" містить унікальний ідентифікатор запиту;

Поле "Вузол-джерело" містить інформацію про вузол, куди необхідно доставити пакет;

Поле "Вузол-одержувач" містить інформацію про вузол, який транслює відповідь на запит маршруту;

Поле "Тайм-аут" містить час життя пакета. За замовчуванням "Тайм-аут" дорівнює 30*Hops секунд;

Поле "Hops" містить бінарне число, яке визначає кількість проміжних вузлів;

У полі "MAC-адреси проміжних вузлів" зберігаються всі MAC-адреси проміжних вузлів.

Коли пакет відповіді на запит маршруту досягає вузла-джерела, вузол записує інформацію з пакета у свій кеш маршрутизації. Крім того, кожен проміжний вузол, через який пройшов пакет, так само зберігає інформацію про маршрут у свій кеш маршрутизації.

Інформація в кеші маршрутизація зберігається у двох таблицях:

- у першій таблиці зберігається вся основна інформація про маршрут, крім проміжних вузлів;
- проміжні вузли зберігаються в другій таблиці, і вона пов'язана з першою первинним ключем.

Структуру кеша маршрутизації представлено в таблиці 2.3 [28].

Таблиця 2.3

Структура даних кеша маршрутизації

16 байт	16 байт	16 байт	8 байт	1 байт
UID	Вузол-джерело	Вузол - одержувач	Тайм-аут	Hops

Поле "UID" містить унікальний ідентифікатор побудованого маршруту;

Поле "Вузол-джерело" містить інформацію про вузол, який ініціював побудову маршруту;

Поле "Вузол-одержувач" містить інформацію про вузол, маршрут до якого було побудовано;

Поле "Тайм-аут" містить час життя маршруту. За замовчуванням дорівнює 300 секунд;

Поле "Hops" містить бінарне число, що дорівнює кількість проміжних вузлів у цьому маршруті.

Інформацію про проміжні вузли наведено в таблиці 2.4.

Таблиця 2.4

Таблиця зберігання інформації про проміжні вузли

16 байт	16 байт
UID	MAC-адреса проміжного вузла
1e2r31we	1.2.3.4
1e2r31we	1.2.3.5
1gd34gas	1.2.3.4

Поле "UID" містить унікальний ідентифікатор побудованого маршруту;

У полі "MAC-адреса проміжного вузла" міститься послідовний запис усіх проміжних вузлів. Кожному проміжному вузлу присвоюється UID маршруту, у якому він перебуває.

Після того, як вузол-джерело отримав запитувану інформацію, він може розпочати передачу даних [29].

2.2. Алгоритм передавання даних

Для передачі даних використовується побудований маршрут між вузлом-джерелом і вузлом-одержувачем. Дані передаються послідовно через кожен проміжний вузол. Після того, як пакет із даними досяг одержувача, одержувач надсилає пакет підтвердження про успішне передання пакета з даними.

У разі, якщо в процесі передавання даних один із проміжних вузлів став недоступним, вузол, який не зміг передати дані за побудованим маршрутом, надсилає пакет із помилкою, у якому вказано недоступний вузол. У цій ситуації вузол, який ініціював передачу даних, починає будувати новий маршрут до одержувача, якщо в кеші маршрутизації не було знайдено альтернативного маршруту.

Блок-схему алгоритму передавання даних представлено на рис. 2.2.

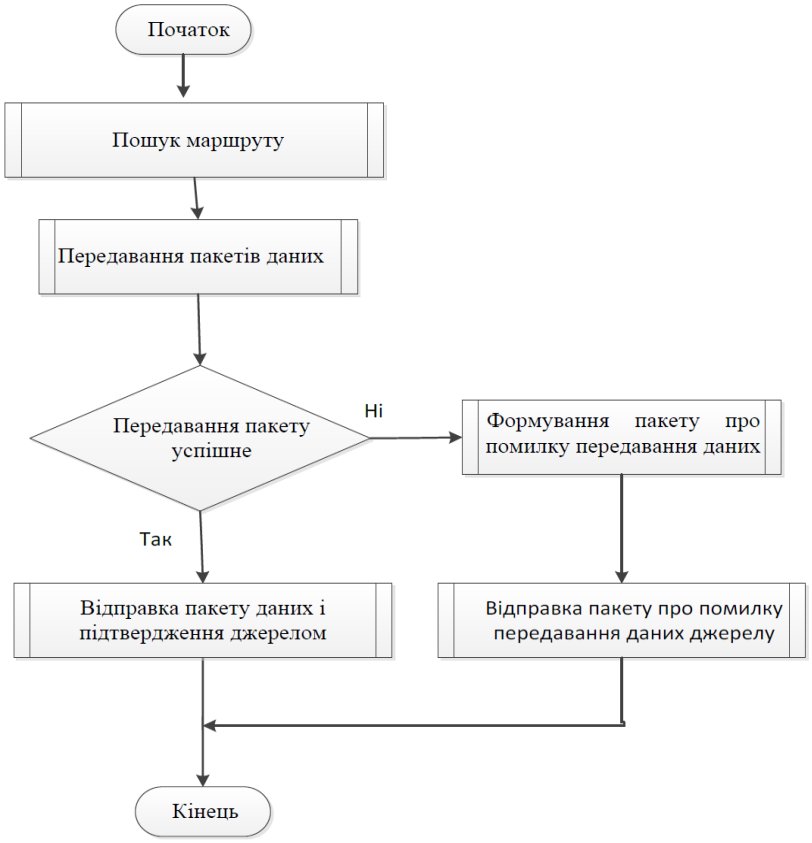

Рис. 2.2. Блок-схема алгоритму передавання даних

Структура пакета передавання даних представлена в таблиці 2.5 [30].

Таблиця 2.5

Структура пакета передавання даних

1 байт	1 байт	16 байт	16 байт	16*N байт	До 65535 байт	1 байт	16 байт
Тип запиту	ID пакету	Вузол-джерело	Вузол-одержувач	MAC-адреси проміжних вузлів	Дані	Останній фрагмент	Розмір вікна

Поле "Тип запиту" містить бінарне число, що відповідає типу пакета. Для пакета передавання даних за замовчуванням 0011;

Поле "ID пакета" містить бінарне число, що ідентифікує номер пакета, для складання пакета підтвердження або помилки доставки пакета;

Поле "Вузол-джерело" містить інформацію про вузол, який ініціював передавання даних;

Поле "Вузол-одержувач" містить інформацію про вузол, куди дані мають бути доставлені;

В поле "MAC-адреси проміжних вузлів" послідовно записано проміжні вузли;

У полі "Дані" знаходиться передана інформація;

В поле "Контрольна сума" містить контрольну суму підраховану для всього пакета цілком;

У полі "Розмір вікна" міститься кількість байт інформації, яку готовий прийняти у свої внутрішні буфери ініціатор передачі даних. Перший пакет, який відправляє джерело, є порожнім, щоб підтвердити те, що маршрут існує, і яку кількість даних він може прийняти за раз: "Розмір вікна".

Ідентифікація типу пакета відбувається за полем "Тип запиту". Ідентифікуючи тип пакета, розпізнається ID пакета, і вузол-джерело визначає, який пакет було доставлено.

Структуру пакета підтвердження про передачу даних наведено в таблиці 2.6.

Таблиця 2.6

Структура пакета підтвердження про передавання даних

1 байт	1 байт	16 байт	16 байт	1 байт	16*N байт	16 байт
Тип запиту	ID пакету	Вузол-джерело	Вузол-одержувач	Hops	MAC-адреси проміжних вузлів	Розмір вікна

Поле "Тип запиту" містить бінарне число, що відповідає типу пакета. Для пакета підтвердження про передавання даних за замовчуванням 0101;

Поле "ID пакета" містить бінарне число, що ідентифікує номер пакета, який було доставлено;

Поле "Вузол-джерело" містить інформацію про вузол, якому необхідно доставити пакет підтвердження;

Поле "Вузол-одержувач" містить інформацію про вузол, куди дані були доставлені;

Поле "Hops" містить бінарне число, що дорівнює кількості проміжних вузлів між джерелом і одержувачем;

В поле "MAC-адреси проміжних вузлів" послідовно записані проміжні вузли;

У полі "Розмір вікна" міститься кількість байт інформації, яку готовий прийняти у свої внутрішні буфери вузол-одержувач.

Структуру пакету про помилку передавання даних наведено в таблиці 2.7.

Таблиця 2.7

Структура пакета про помилку передавання даних

1 байт	1 байт	16 байт	16 байт	1 байт	16*N байт	16 байт
Тип запиту	ID пакету	Вузол-джерело	Вузол-одержувач	Hops	MAC-адреси проміжних вузлів	MAC-адреса вузла, на якому стався збій

Поле "Тип запиту" містить бінарне число, що відповідає типу пакета. Для пакета про помилку передавання даних за замовчуванням 0110;

Поле "ID пакета" містить бінарне число, що ідентифікує номер пакета, який не був доставлений;

Поле "Вузол-джерело" містить інформацію про вузол, куди необхідно доставити пакет про помилку передавання даних;

Поле "Вузол-одержувач" містить інформацію про вузол, маршрут до якого було порушено;

Поле "Hops" містить бінарне число, що дорівнює кількості проміжних вузлів між джерелом і одержувачем;

У полі "MAC-адреси проміжних вузлів" послідовно записані проміжні вузли;

У полі "MAC-адреса вузла, на якому стався збій" міститься інформація про вузол, до якого пакет передавання даних відправити не вдалося.

2.3. Алгоритм фрагментації великих пакетів

Для передавання великих даних необхідно розбивати їх на пакети і передавати у вигляді дейтаграм, довжина яких може в підсумку зможе перевищити максимально можливу кількість даних, які може прийняти абонент. За рахунок фрагментування - розбиття "великого пакета" на деяку кількість частин (фрагментів), розмір кожної частини задовольнить вимоги абонента. Після того, як усі фрагменти будуть передані через проміжну мережу, вони будуть зібрані на вузлі-одержувачі модулем "збірки у великий пакет". Збірку пакета з фрагментів здійснює тільки одержувач, а не будь-який з абонентів. Проміжні абоненти можуть тільки фрагментувати пакети, але не збирати їх [25-28].

Для того, щоб не переплутати фрагменти різних пакетів, використовується поле "ID пакета", значення якого має бути однаковим для всіх фрагментів одного пакета та не повторюватися для різних пакетів, доки в обох пакетів не закінчився час життя. Під час поділу даних пакета, розмір усіх фрагментів, крім останнього, має бути кратний 8 байтам. Це дає змогу відвести менше місця в заголовку під поле Зміщення фрагмента.

Другий біт поля Прапори (Останній фрагмент), якщо дорівнює одиниці, вказує на те, що цей фрагмент - не останній у пакеті. Якщо пакет відправляється без фрагментації, прапор "Останній фрагмент" встановлюється в 0.

У результаті проєктування було отримано:

- Алгоритм пошуку маршруту;
- алгоритм передачі даних;
- алгоритм фрагментації даних.

Крім того, було спроєктовано структури даних:

- структура пакета даних запиту маршруту;
- структура пакета даних відповіді на запит;
- структура даних у кеші;
- структура пакета даних про проміжні вузли;
- структура пакета передавання даних;
- структура пакета даних підтвердження про передачу даних;
- структура пакета даних про помилку передавання даних.

РОЗДІЛ 3
АЛГОРИТМІЧНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПЕРЕДАЧІ ДАНИХ ЧЕРЕЗ БЕЗДРОТОВІ ДЕЦЕНТРАЛІЗОВАНІ МЕРЕЖІ

3.1. Аналіз мобільних операційних систем

Чи знали ви, що Android від Google є найпопулярнішою мобільною операційною системою в усьому світі? Однак iOS від Apple панує у Сполучених Штатах. У 4 кварталі 2023 року Android посідає перше місце з часткою ринку 69,44% у всьому світі, випереджаючи iOS з різницею в 39,55%, яка має частку ринку 29,89%. Тим часом у США Apple продовжує домінувати з часткою ринку 57,11%, випереджаючи Android з різницею в 14,70%.

Давайте заглибимося у ринкову частку двох основних операційних систем наших пристроїв - iOS, на якій працюють iPhone, та Android, на якій працюють багато телефонів, вироблених багатьма компаніями.

У четвертому кварталі 2023 року Android залишалася найпопулярнішою операційною системою у світі з часткою ринку 69,44%, за нею слідував її головний конкурент, iOS, з 29,89%. Обидві операційні системи складають понад 99% частки ринку операційних систем у світі [38].

Android - це платформа з відкритим вихідним кодом, що належить Google, яка може бути адаптована і використовуватися багатьма брендами смартфонів. Це одна з причин, чому Android керує ринком мобільних операційних систем. Тим часом, iOS розроблена лише для продуктів Apple.

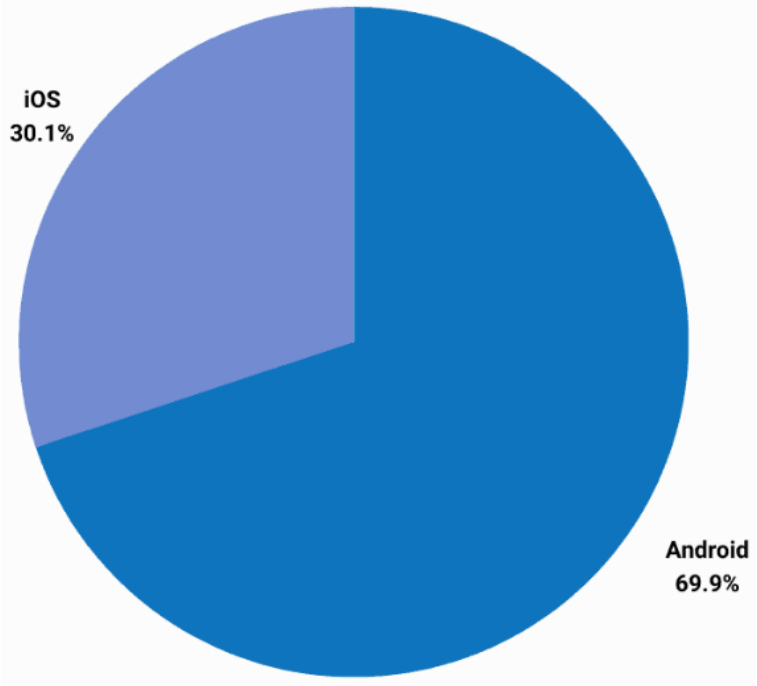

Рис. 3.1. Розподіл популярності операційних систем

Android вийшла на перше місце на світовому ринку у 2012 році, випередивши iOS на 3,37%. Відтоді вона продовжує домінувати на ринку мобільних операційних систем. З 2009 року по теперішній час Android зросла більш ніж на 67%, в той час як частка ринку iOS знизилася приблизно на 5% [38].

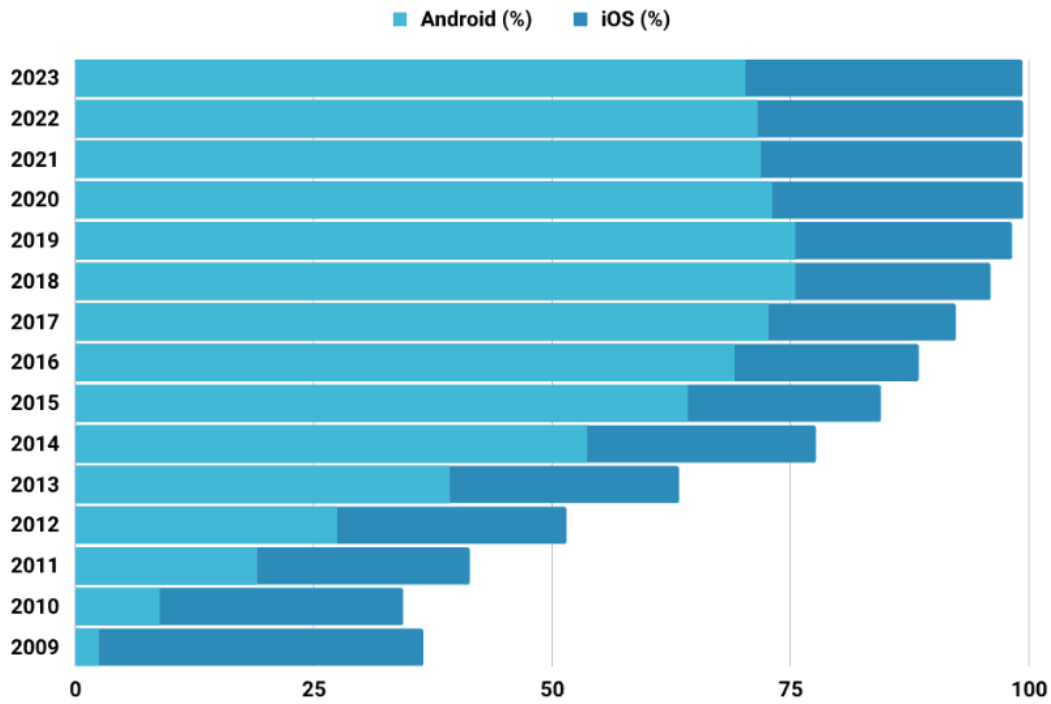

Рис. 3.2. Розподіл популярності операційних систем за роками

Проведемо порівняльний аналіз Android та iOS систем, за функціональними характеристиками, що мають значення в розробці цієї системи.

Кожна із систем має як низку суттєвих переваг, так і недоліків. Далі наведено основні, явно виділені плюси та мінуси кожної системи.

Операційна система Android. Переваги операційної системи Android [38]:

- простота і гнучкість користувацького інтерфейсу;
- відкритість операційної системи;
- невисока ціна на мобільні пристрої;
- висока частка на ринку;
- легка і безкоштовна публікація застосунків. Недоліки операційної системи Android:
- високе енергоспоживання;
- низька захищеність.

Операційна система iOS. Переваги операційної системи iOS:

- простота користувацького інтерфейсу;
- низьке енергоспоживання;
- висока захищеність;
- найбільш швидкий відгук системи при роботі з зовнішнім інтерфейсом;

Недоліки операційної системи iOS:

- висока ціна на мобільні пристрої.

Висновки з аналізу представлених мобільних ОС. Проаналізувавши переваги та недоліки цих операційних систем, можна зробити висновок, що оптимальним варіантом для розробки застосунку буде OS Android. Android має значну частку на ринку, доступну ціну і лише трохи поступається iOS у плані роботи з інтерфейсом.

3.2. Аналіз програмних засобів розробки мобільних додатків

Середовище розробки Eclipse. Eclipse - вільне інтегроване середовище розроблення модульних кросплатформених додатків. Розвивається і підтримується Eclipse Foundation.

Eclipse є безкоштовною програмною платформою з відкритим вихідним кодом, контролюється організацією Eclipse Foundation. Eclipse написана на мові програмування Java і основною метою її створення є підвищення продуктивності процесу розробки програмного забезпечення.

Претендує на статус найпопулярнішої Java IDE. Особливості платформи Eclipse [31]:

- кросплатформеність - працює під операційними системами Windows, Linux, Solaris і Mac OS X;
- Використовуючи Eclipse, можна програмувати багатьма мовами, як-от Java, C і C++, PHP, Perl, Python, Cobol та іншими;
- Eclipse є інструментом для створення практично будь-якого клієнтського програмного забезпечення [33].

Середовище розробки Android Studio. Android Studio - інтегроване середовище розробки додатків для операційної системи android [31-37], покликане замінити собою Eclipse. Остання, на даний момент є найпопулярнішим інструментом у колі Android розробників. Процес розробки в середовищі Android Studio став ще більш гнучким ніж у головного конкурента. Це досягається за рахунок відображення всіх робочих файлів прямо в структурі проєкту. Вкрай корисною є можливість бачити всі візуальні зміни проєкту в режимі реального часу. Ще ця SDK дає можливість протестувати роботу майбутнього додатка на різних пристроях. У вбудованому емуляторі Android девайсів можна встановити різноманітні технічні конфігурації та роздільну здатність екрана для тесту. Крім того, при використанні емуляції є можливість отримати інформацію про приблизний рівень продуктивності для того чи іншого пристрою.

Android SDK - середовище розробки додатків для операційної системи Android. Воно дає змогу створювати й тестувати Android-додатки, що використовують камеру мобільного пристрою, акселерометр, компас, дані GPS, доступ через Bluetooth, Wi-Fi, EDGE і 3G. Android SDK містить різноманітні бібліотеки, документацію та інструменти, які допомагають розробляти мобільні додатки для платформи Android [35].

API Android SDK - API бібліотеки (це скорочення від Application Programming Interface) Android, що надаються для розробки додатків. Документація SDK - включає велику довідкову інформацію, що деталізує, що включено в кожен пакет і клас як це використовувати під час розроблення застосунків [34].

AVD (Android Virtual Device) - інтерактивний емулятор мобільного пристрою Android. Використовуючи емулятор, можна запускати і тестувати додатки без використання реального Android пристрою [36].

Development Tools - SDK включає кілька інструментальних засобів для розробки, які дають змогу компілювати та налагоджувати створювані додатки.

Sample Code - Android SDK надає типові додатки, які демонструють деякі з можливостей Android, і прості програми, які показують, як використовувати індивідуальні особливості API в коді.

З іншого боку Android Studio споживає багато пам'яті і тому, можливі деякі проблеми з установкою, а так само і роботою. Якщо ПК не відповідає певним параметрам, швидкість роботи за Android Studio може бути рази в 3-5 нижчою, ніж в інших середовищах. Так само, це середовище часто оновлюється, часом, оновлення бувають через кілька днів, тобто можливо й інтерфейс постійно змінюється, що може ускладнювати роботу недосвідченого розробника.

До того ж без цього пакета будь-які розробки просто неможливі, в якому б середовищі програмування вони не проводилися [33].

Порівняльний аналіз Eclipse і Android Studio. Eclipse є найбільш усталеним продуктом і користується величезною популярністю серед користувачів, які давно про-

грамують, особливо мовою java. Крім того, Google надав ADT плагін, який перетворює популярний IDE на повнофункціональне середовище розробки під Android. У зв'язку з цим, люди не поспішають переходити в Android Studio.

У зв'язку з тим, що обидва середовища розробки є багатообіцяючими і користуються величезною популярністю, не поступаючись одне одному, порівняємо їх у 5 різних галузях.

Системи автоматичного складання проєкту. Android Studio використовує систему збірки Gradle. Він ґрунтується на Apache Ant і Apache Maven, але він також містить Groovy DSL (Domain-Specific Language), що дає змогу сценарію збірки, який відкриває багато можливостей автоматизації, як-от завантаження бети .apk у TestFlight для тестування. Eclipse, з іншого боку, використовує Apache Ant як основну систему збірки, яка дуже надійна, і люди з нею давно знайомі [37].

Написання коду. Обидва середовища розробки мають стандартний код Java, але у випадку Android Studio, Google створив найбільш вдосконалену систему перевірки коду. Android Studio може реорганізувати свій код у тих місцях, де це просто не можливо за допомогою Eclipse і ADT. Тобто, Android Studio може краще передбачити те, що хоче зробити користувач, ніж середовище Eclipse.

Інтерфейс середовища розробки. Однією з основних особливостей створеної Google в Android Studio, є повністю перероблений користувацький інтерфейс. Інтерфейс в Android Studio має більше варіантів налаштування і робиться все це за допомогою спеціальних інструментів, що в Eclipse користувач повинен був вибирати вручну в XML.

Створення та організація проєкту. Обидва середовища розробки працюють по різному під час організації проєкту. Щоб використовувати Eclipse, то необхідні знання та вміння роботи в робочих просторах. Під час запуску Eclipse, необхідно вибрати робочий простір, у якому зберігаються проєкти, і потім завантажити його в дерево процесів. Основним мінусом цього підходу є те, що для того щоб переключитися на проєкт в іншому робочому просторі, необхідно повністю перезавантажити середовище розробки. Android Studio розглядає цю ситуацію інакше, вводячи поняття моду-

лів. Під час роботи з додатком може бути відкритий будь-який модуль, потрібна бібліотека класів, яку завантажили за допомогою SDK, а так само можуть бути запущені засоби інтеграції. Кожен з цих модулів може мати свої власні Gradle збірки і мати свої окремі залежності.

Продуктивність і стабільність. Eclpise є середовищем розробки, заснованим на java. Щоб середовище розробки працювало надійно, необхідна пристойна кількість оперативної пам'яті та хороша потужність процесора. Для Eclipse не є незвичним те, що стався збій під час експорту або імпорту APK. Android Studio є більш надійним середовищем розробки, в ній рідко відбуваються непередбачені збої, а якщо такі є, то користувачеві буде показано відповідний код помилки. Проте, Android Studio вимагає високих обчислювальних потужностей, за рахунок багатого вбудованого інструментарію [31-37].

3.3. Розробка модуля пошуку сусідніх вузлів

Опис використаного API. Wi-Fi P2P дає змогу застосунку швидко знаходити та взаємодіяти з прилеглими пристроями, що перебувають за межами можливостей Bluetooth.

API-інтерфейси Wi-Fi Peer-to-peer (P2P) дають змогу застосунку під'єднуватися до сусідніх пристроїв без необхідності під'єднання до мережі або точки, і якщо застосунок призначений для роботи в захищеній мережі, Wi-Fi Direct є більш підходящим варіантом, ніж традиційні мережі Wi-Fi для мереж ad-hoc, з таких причин [27]:

- Wi-Fi Direct підтримує шифрування WPA2. (Деякі мережі ad-hoc підтримують тільки WEP-шифрування).
- Пристрої можуть транслювати послуги, які вони надають, що допомагає іншим пристроям виявляти відповідні однорангові вузли у найпростіший спосіб.

Крім того, технологія Wi-Fi Direct має вбудовану можливість керування живленням, користувацьким інтерфейсом і сервісом кожного пристрою, і цю інформацію можна використовувати для вибору пристрою, який може найефективніше працювати в ролі сервера.

API-інтерфейси Wi-Fi P2P складаються з таких основних частин [27]:

- Методи, які дають змогу виявляти, запитувати і підключатися до однорангових вузлів, визначені в класі WifiP2pManager;
- Методи, які дають змогу отримувати повідомлення про успішне або невдалому виклику класу WifiP2pManager. Під час виклику WifiP2pManager кожен метод може отримати певний радіосигнал, переданий як параметр.
- Сигнали, які повідомляють про конкретні події, виявлені фреймворком Wi-Fi P2P, наприклад, про скинуте з'єднання або недавно виявлений сусідній вузол.

Найчастіше використовуються три основні компоненти API. За допомогою WifiP2pManager.ActionListener можна викликати метод discoverPeers(), який шукає найближчих сусідів, і за допомогою методів ActionListener.onSuccess() і ActionListener.onFailure() отримувати необхідні повідомлення. WIFI_P2P_PEERS_CHANGED_ACTION - так само транслюється, якщо метод discoverPeers() виявляє, що список сусідніх вузлів змінився.

Клас WifiP2pManager надає методи, що дають змогу взаємодіяти з апаратним забезпеченням Wi-Fi пристрою, щоб виконувати такі дії, як виявлення сусідніх вузлів і під'єднання до них. Доступні такі методи:

- initialize() – реєструє додаток у фреймворку Wi-Fi. Цей метод необхідно викликати до виклику будь-якого іншого методу Wi-Fi P2P;
- connect() – запускає однорангове з'єднання з пристроєм із зазначеною конфігурацією;
- cancelConnect() – скасовує будь-які поточні узгодження групи однорангових груп;
- requestConnectInfo() – запит інформації про підключення пристрою;

- createGroup() – створює однорангову групу з поточним пристроєм як власником групи;
- requestGroupInfo() – запит інформації групи однорангової мережі;
- discoverPeers() – ініціює однорангове виявлення;
- requestPeers() – запитує поточний список виявлених однорангових вузлів.

Методи WifiP2pManager дозволяють передавати пристрою інформацію про стан підключення, щоб фреймворк Wi-Fi P2P міг повідомляти користувача про статус виклику. Основні інтерфейси, що використовуються WiFiP2pManager, показані в таблиці 3.1:

Таблиця 3.1

Інтерфейси, що використовуються API WiFiP2pManager

Інтерфейс	**Пов'язані методи**
WifiP2pManager.ActionListener	*connect(), cancelConnect(), createGroup(), removeGroup(), discoverPeers()*
WifiP2pManager.ChannelListener	*initialize()*
WifiP2pManager.ConnectionInfoListener	*requestConnectInfo()*
WifiP2pManager.GroupInfoListener	*requestGroupInfo()*
WifiP2pManager.PeerListListener	*requestPeers()*

API-інтерфейси Wi-Fi P2P визначають сигнали, які передають у разі виникнення певних подій Wi-Fi P2P, наприклад, у разі виявлення нового вузла або зміни стану Wi-Fi пристрою [23].

Сигнали та їхні значення описані в таблиці 3.2.

Таблиця 3.2

Види та значення сигналів

Сигнал	**Опис**
WIFI_P2P_CONNECTION_CHANGED_ACTION	Широкомовна передача про зміну стану підключення Wi-Fi пристрою
WIFI_P2P_PEERS_CHANGED_ACTION	Мовлення, використовується метод discoverPeers(). Зазвичай надсилається запит requestPeers(), щоб отримати оновлений список однорангових вузлів
WIFI_P2P_STATE_CHANGED_ACTION	Широкомовна передача під час увімкнення або вимкнення Wi-Fi P2P на пристрої
WIFI_P2P_THIS_DEVICE_CHANGED_ACTION	Сигнал, коли деталі пристрою змінилися, наприклад, ім'я пристрою

Створення широкомовного приймача. Широкомовний приймач дає змогу отримувати сигнали, які транслює система Android, щоб застосунок міг реагувати на події, що цікавлять користувача. Основні кроки для створення широкомовного приймача для керування налаштуваннями Wi-Fi P2P полягають у такому:

- Необхідно створити клас, який успадковує клас BroadcastReceiver. Для конструктора класу необхідні параметри класу WifiP2pManager, WifiP2pManager.Channel і повідомлення про те, що широкомовний приймач був зареєстрований. Це дає змогу за допомогою широкомовного каналу оновлювати інформацію, а так само задіяти апаратне забезпечення і канал зв'язку, якщо це необхідно;
- Метод onReceive() дає змогу вказати сигнали, які нас цікавлять, і які дії необхідно виконати під час отримання того чи іншого сигналу. Наприклад, якщо широкомовний приймач отримує намір WIFI_P2P_PEERS_CHANGED_ACTION, можна викликати метод requestPeers(), щоб отримати список виявлених у цей момент вузлів.

На рис. 3.3 показано код, на якому можна побачити дії додатка під час отримання певного типу сигналу:

```
>>> public class WiFiDirectBroadcastReceiver extends BroadcastReceiver {
... private WifiP2pHanager mHanager;
... private Channel mChannel;
... private MyWiFiActivity mActivity;
... public WiFiDirectBroadcastReceiver(WifiP2pManager manager, Channel channel, M
    yWifiActivity activity) {
... super();
... this.mHanager = manager;
... this.mChannel = channel;
... this.mActivity = activity;
... }
... (©Override
... public void onReceive(Context context. Intent intent) { String action = inten
    t.getAction();
... if (WifiP2pHanager.WIFI_P2P_STATE_CHANGED_ACTION.equals(action)) {
... // Check to see if Wi-Fi is enabled and notify appropriate activity
... } else if (WifiP2pManager.WIFI_P2P_PEERS_CHANGED_ACTI0N.equals(action)) {
... // Call WifiP2pHanager. requestPeers () to get a list of current peers
... } else if (WifiP2pHanager.WIFI_P2P_C0NNECTI0N_CHANGED_ACTI0N.equals(action))
    { // Respond to new connection or disconnections
... } else if (WifiP2pHanager.WIFI_P2P_THIS_DEVICE_CHANGED_ACTION.equals(action))
    { // Respond to this device's wifi state changing
... }
... }
```

Рис. 3.3. Опис дій програми в разі отримання певного типу сигналу

На рис. 3.3. показано приклад реалізації методу onReceive() з чотирма видами сигналу: перевірка можливості використання Wi-Fi, запит списку доступних вузлів, сигнал про нове під'єднання або від'єднання від пристрою, відповідь на зміни стану в мережі.

Налаштування дозволів у маніфесті. Для того, щоб використовувати Wi-Fi P2P, необхідно додати CHANGE_WIFI_STATE, ACCESS_WIFI_STATE і INTERNET у Android Manifest. Wi-Fi P2P не вимагає підключення до Інтернету, але використовує стандартні сокети Java, для яких INTERNET вимагає дозволу. Таким чином, для використання Wi-Fi P2P необхідні дозволи, показані на рис. 3.4:

```
>>> <uses-permission android:name="android,permission.ACCESS_WIFI_STATE" />
... <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
... <uses-permission android:name="android.permission.CHANGE_NETriORK_STATE" />
... Cuses-permission android:name="android.permission.INTERNET" />
```

Рис. 3.4. Список необхідних дозволів

На рис. 3.4 показано список дозволів для використання функцій пристрою: доступ до Wi-Fi і зміни його стану, до мережі та інтернету.

Налаштування широкомовного приймача. Щоб використовувати Wi-Fi P2P, необхідно прослухати трансляції, які повідомляють додатку, коли відбуваються певні події. У застосунку було створено екземпляр і встановлено IntentFilter для отримання такої інформації:

- -WIFI_P2P_STATE_CHANGED_ACTION - вказує, чи ввімкнено Wi-Fi P2P;
- WIFI_P2P_PEERS_CHANGED_ACTION - вказує, що список доступних однорангових вузлів змінився;
- WIFI_P2P_CONNECTION_CHANGED_ACTION - показує стан підключення Wi-Fi P2P;
- WIFI_P2P_THIS_DEVICE_CHANGED_ACTION - вказує, що дані конфігурації пристрою змінилися [24].

Приклад коду, що реалізує отримання повідомлень, зазначених вище, показано на рис. 3.5.

```
>>> private final IntentFilter intentFilter = new IntentFilter();
... private Channel channel;
... private BrcadcastReceiver receiver = null;
... / * *
... * gparam isWifiP2pEnabLed the isWifiP2pEnabled to set
... */
... public void setIsWifiP2pEnabled (boolean isWifiP2pEnabled)
... { this.isWifiP2pEnabled = isWif iP2pEnabled;
... }
... @Override
... public void onCreate(Bundle savedlnstanceState) { super.onCreate(savedlnstanceState
    );
... setContentView (R. layout .main) ;
... // add necessary intent values to be matched.
... intentFilter. addAction (WifiP2pManager. WIFI_P2P_STATE_CHANGED_ACTION) ;
... intentFilter. addAction (WifiP2pManager. WIFI_P2P_PEERS_CHANGED_ACTION) ;
... intentFilter. addAction (WifiP2pManager. WIFI_P2P_CONNECTION_CHANGED_ACTION) ;
... intentFilter. addAction (Wif iP2pManager. WIFI_P2P_THIS_DEVICE_CHANGED_ACTION) ;
```

Рис. 3.5. Методи класу IntentFilter

У результаті методу onCreate() отримується екземпляр об'єкта WifiP2pManager і його можна викликати методом initialize(). Метод WifiP2pManager.Channel повертає

об'єкт, який використовується для підключення додатка до фреймворку Wi-Fi P2P. Приклад коду показано на рис. 3.6.

```
>>> Channel mChannel;
... public void onCreate(Bundle savedlnstanceState) {
... mManager = (WifiP2pManager) getSystemService(Context.WIFI_P2P_SERVICE);
... mChannel = mManager.initialize(this, getMainLooper()} null);
```

Рис. 3.6. Використання методу OnCreate()

Тепер створимо новий клас BroadcastReceiver, який будемо використовувати для прослуховування змін у стані системи Wi-Fi P2P. У методі onReceive() додамо умови для опрацювання кожної зміни стану P2P, зазначеного вище. Результат отриманого коду показано на рис. 3.7.

```
>>> @Override
... public void cnReceive(Context context, Intent intent)
... { String action = intent.getAction();
... if (WifiP2pManager. WIFI_P2P_STATE_CHANGED_ACTION.equals (action)) {
... // UI update to indicate vifi p2p status.
... int state = intent.getlntExtra(WifiP2pManager.EXTRA_WIFI_STATE, -1);
... if (state = WifiP2pManager. WIFI_P2P_STATE_ENABLED) {
...     // Wifi Direct mode is enabled
... activity.setlsWifiP2pEnabled(true);
... } else {
... activity.setIsWifiP2pEnabled(false); activity.resetData();
... l
... Log.d(WiFiDirectActivity.TAG, "P2P state changed - " + state);
... } else if (Wifi P2pManager.WIFI_P2P_PEERS_CHANGED_ACTI0N.equals(action)) {
... // request available peers from the vifi p2p manager. This is an
... // asynchronous call and the calling activity is notified vith a
... // callback on PeerListListener.onPeersAvailable()
... if (manager != null) {
... manager.requestPeers(channel, (PeerListListener) activity.getFragmentManager()
...                     .findFragmentById(R. id. frag_LLst));
... }
... Log. d( WiFi Di re ctActivi ty. TAG, "P2P peers changed");
... } else if (WifiP2pManager.WIFI_P2P_C0NNECTI0N_CHANGED_ACTI0N.equals (action)) {
... if (manager == null) { return:
... }
... Networkinfo networkinfo = (Networkinfo)
... intent.getParcelableExtra (WifiP2pManager.EXTRA_NETWORK_INFO);
... if (networkinfo.isConnected()) {
... // ve are connected vith the other device, request connection
... // info to find group ovner IP
... DeviceDetailFragment fragment = (DeviceDetailFragment) activity.getFragmentManager()
    .findFragmentById(R.id.frag_detail);
... manager.requestConnectionlnfo(channel, fragment);
... } else {
... // It's a disconnect activity.resetData(); }
... } else if (WifiP2pManager. WIFI_P2P_THIS_DEVICE_CHANGED_ACTION.equals (action)) {
...     DeviceListFragment fragment = (DeviceListFragment) activity.getFragmentManager()
... .findFragmentByld(R.id.frag_List);
... fragment.updateThisDevice ((WifiP2pDevice) intent.getParcelableExtra (
... WifiP2pManager.EXTRA_WIFI_P2P_DEVICE) );
```

Рис. 3.7. Детальний опис дій у методі OnReceive()

Необхідно встановити IntentFilter і BroadcastReceiver у стан паузи, коли активний основний потік. Найкраще місце, щоб зробити це - onResume() і onPause() методи, код яких показано на рис. 3.8.

```
>>> @Override
... public void onResume() {
... super.onResume();
... receiver = new WiFiDirectBroadcastReceiver (nVIanager, mChannel, this);
... }
... registerReceiver(receiver, intentFilter);
... @Override
... public void onPause();
... {super.onPause();
... unregisterReceiver(receiver);
... }
```

Рис. 3.8. Методи OnResume() та OnPause()

Щоб виявити сусідні вузли, до яких можна підключитися, необхідно викликати функцію discoverPeers() для виявлення доступних однорангових вузлів, що знаходяться в зоні дії. Виклик цієї функції є асинхронним, і успіх або відмову передають у застосунок за допомогою методів onSuccess() і onFailure(), що належать до класу WifiP2pManager.ActionListener. Метод onSuccess() повідомляє про те, що процес відкриття вдався і не дає жодної інформації про реальні вузли, які він виявив, якщо такі є. Приклад коду показано на рис. 3.9.

```
>>> @Override
... public void connect(WifiP2pConfig config) {
... manager.connect(channel, config, new ActionListener() {
...
... @Override
... public void onSuccess() {
... // WiFiDirectBroadcastReceiver will notify us.Ignore for now. }
...
... @Override
... public void onFailure(int reason) { Toast.makeText(WiFiDirectActivity.this,
    "Помилка підключення". Toast. LENGTH_SHORT) .show () ;
... }
... });
... }
```

Рис. 3.9. Приклад методу OnSuccess() і OnFailure()

Розробка модуля побудови списку абонентів. Якщо процес виявлення завершується успішно і сусідні вузли виявлені, система передає сигнал

WIFI_P2P_PEERS_CHANGED_ACTION, який можна використовувати для отримання списку сусідніх вузлів. Коли додаток отримує сигнал WIFI_P2P_PEERS_CHANGED_ACTION, можна запросити список виявлених одноранговых вузлів за допомогою requestPeers(). Приклад коду показано на рис. 3.10:

```
>>> if (manager != null) { manager.requestPeers(channel, (PeerListListener) acti
    vity.getFragmentManager()
... . findFragmentByld (R.id.fra.g_Llst));
... }
... Log.d(WiFiDirectActivity. TAG, "P2P peers changed");
... } else if (WifiP2pManager. WIFI_P2P_CONNECTION_CHANGED_ACTI0N. equals (actio
    n)) {
... if (manager — null) { return;
... }
```

Рис. 3.10. Приклад реалізації методу RequestPeers()

Метод requestPeers() асинхронний і може надсилати сповіщення, коли список пірів доступний за допомогою onPeersAvailable(), який визначено в інтерфейсі WifiP2pManager.PeerListListener. Метод onPeersAvailable() надає параметр WifiP2pDeviceList, який можна використовувати для вибору пристрою, до якого необхідно підключитися.

Коли визначили пристрій, до якого необхідно під'єднатися, зі списку отриманих сусідніх вузлів, викличемо метод connect() для під'єднання до пристрою. Для виклику цього методу WifiP2pConfig потрібен об'єкт, що містить інформацію про пристрій, до якого потрібно підключитися. За допомогою WifiP2pManager.ActionListener можна отримувати повідомлення про успішне або невдале підключення до пристрою. У коді на рис. 3.11 показано підключення до потрібного пристрою:

```
>>> @Override
... public void connect(WifiP2pConfig config) {
... manager.connect(channel, config, new ActionListener() {
...
... @Override public void onSuccess() {
... }
...
... @Override
... public void onFailure(int reason) {
... Toast.makeText (WiFiDirectActivity.this, "Помилка підключення"
... Toast.LENGTH_SHORT) .show () ;
... }
... });
... }
```

Рис. 3.11. Приклад використання методу connect()

На рис. 3.11 показано підключення до пристрою зі списку доступних вузлів і виведення помилки, якщо підключення не вдалося.

3.4. Розробка модуля передавання даних

Після встановлення з'єднання можна передавати дані між пристроями за допомогою сокетів. Основні кроки передачі даних такі:

- Створити ServerSocket. Цей сокет очікує підключення від клієнта на зазначеному порту і блокує його доти, доки це не відбудеться. Процес працює у фоновому потоці;
- Створити ClientSocket. Клієнт використовує порт серверного сокета для підключення до серверного пристрою;
- Надсилати дані від клієнта на сервер. Коли клієнтський сокет успішно підключається до серверного сокета, можна надсилати дані з клієнта на сервер;
- Сокет сервера очікує підключення клієнта за допомогою методу accept(). Цей виклик блокується доти, доки клієнт не підключиться, тому цей виклик відбувається в іншому потоці. Коли відбувається з'єднання, серверний пристрій може отримувати дані від клієнта. З файлами можна виконувати будь-які дії: зчитувати, відкривати тощо.

```
>>> if (manager != null) {
... manager.requestpeers(channel, (PeerListListener) activity.getFragmentManager()
    .findFragmentBy!d(R.id.frag_List));
... }
... Log.dfWiFiDirectActivity.TAG, "P2P peers changed");
... } else if (WifiP2pManager.WIFI_P2P_CONNECTION_CHANGED_ACTION.equals (action))
    {
... if (manager == null) { return;
... }
```

Рис. 3.12. Приклад коду для створення сокета сервера

На рис. 3.12 показано код, який дає змогу надсилати файли користувача іншому користувачеві, до якого наразі під'єднаний застосунок, передавання й отримання даних відбувається у фоновому режимі. На клієнті необхідно під'єднатися до серверного сокета для можливості передавання даних. Тільки коли буде встановлено з'єднання, шляхом підтвердження з'єднання з боку обох пристроїв, відкриється можливість передачі даних.

```
>>> @Override
... public void connect(WifiP2pConfig config) {
... manager.connect(channel, config, new ActionListener() {
...
... @Override
... public void onSuccessf) {
...
... @Override
... public void onFailure(int reason) {
... Toast.makeText(WiFiDirectActivity.this, "Помилка підключення"
... Toast.LENGTH_SHORT) .show();
... });
... }
```

Рис. 3.13. Приклад коду для роботи клієнта

У прикладі, показаному на рис. 3.13, показано методи класу FileTransferService: надсилання і вибір файлу, визначення джерела та одержувача.

У результаті розробки було реалізовано: алгоритм пошуку маршруту, алгоритм передавання даних.

Було створено сервіси для роботи пристроїв у вигляді клієнта і сервера, привласнюючи їх коли їм це необхідно.

РОЗДІЛ 4
ТЕСТУВАННЯ РОЗРОБЛЕНОГО ДОДАТКУ НА ANDROID ПРИСТРОЇ

Для тестування програми було використано мобільний пристрій Samsung Galaxy S23 ULTRA (SM-S918BZKHSEK). На рис. 4.1 показано головний екран програми.

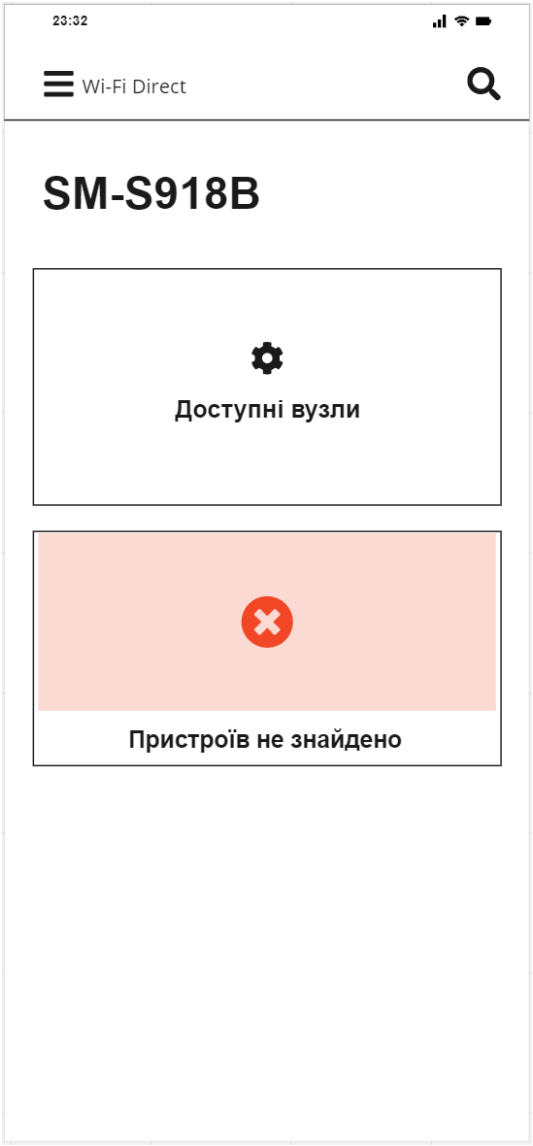

Рис. 4.1. Головний екран застосунку

У фрагменті "Я" показано найменування пристрою, на якому використовується застосунок. У мене це пристій SM-S918B.

У фрагменті "Доступні вузли" показано напис "Пристроїв не знайдено", оскільки Wi-Fi Direct ще не був активований.

У фрагменті "Wi-Fi Direct" є дві кнопки меню, перша відкриває налаштування для ввімкнення Wi-Fi, друга для ініціалізації пошуку доступних вузлів.

Натиснувши на значок у вигляді "Лупи", ми ініціюємо пошук доступних вузлів. На рис. 4.2 показано процес пошуку доступних вузлів.

Рис. 4.2. Пошук доступних вузлів

Після того, як додаток виявить сусідній пристрій, у фрагменті "Доступні вузли" відобразяться знайдені пристрої. Результат показано на рис. 4.3.

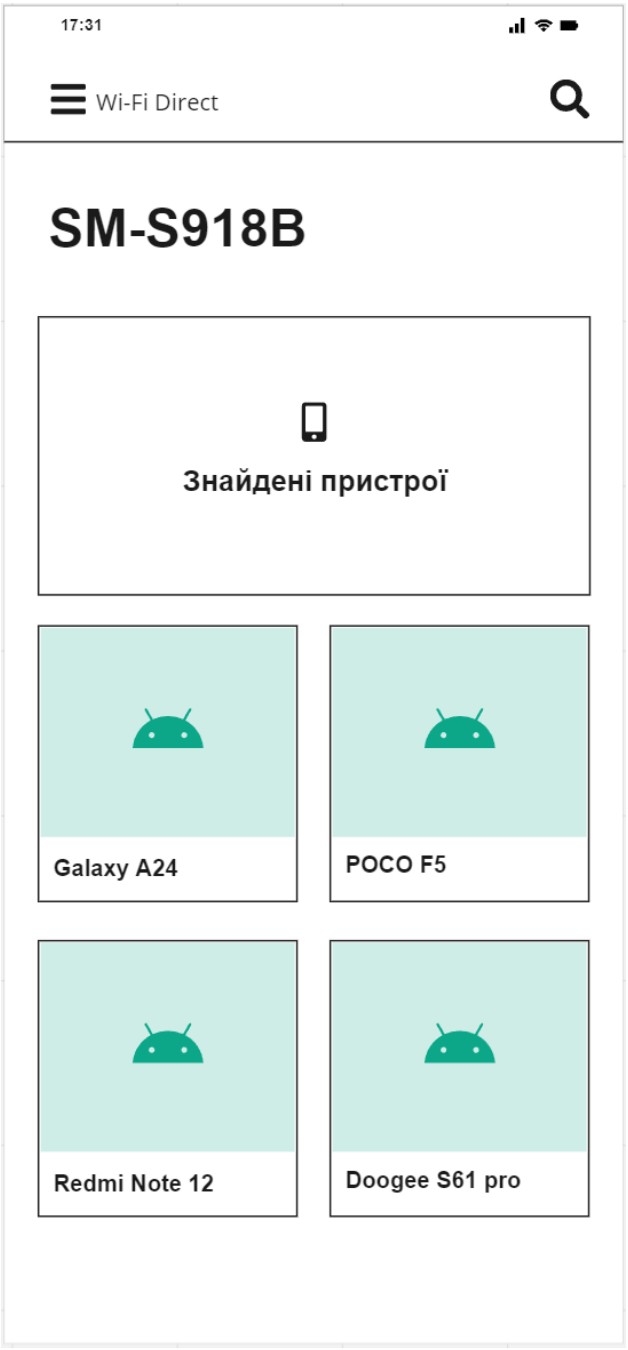

Рис. 4.3. Результат пошуку доступних вузлів

У фрагменті "Доступні вузли" відображені знайдені сусіди, які транслюють сигнал за допомогою Wi-Fi Direct.

При одному натисканні на один з доступних пристроїв, ми можемо отримати детальну інформацію про пристрій. Інформація показана на рис. 4.4.

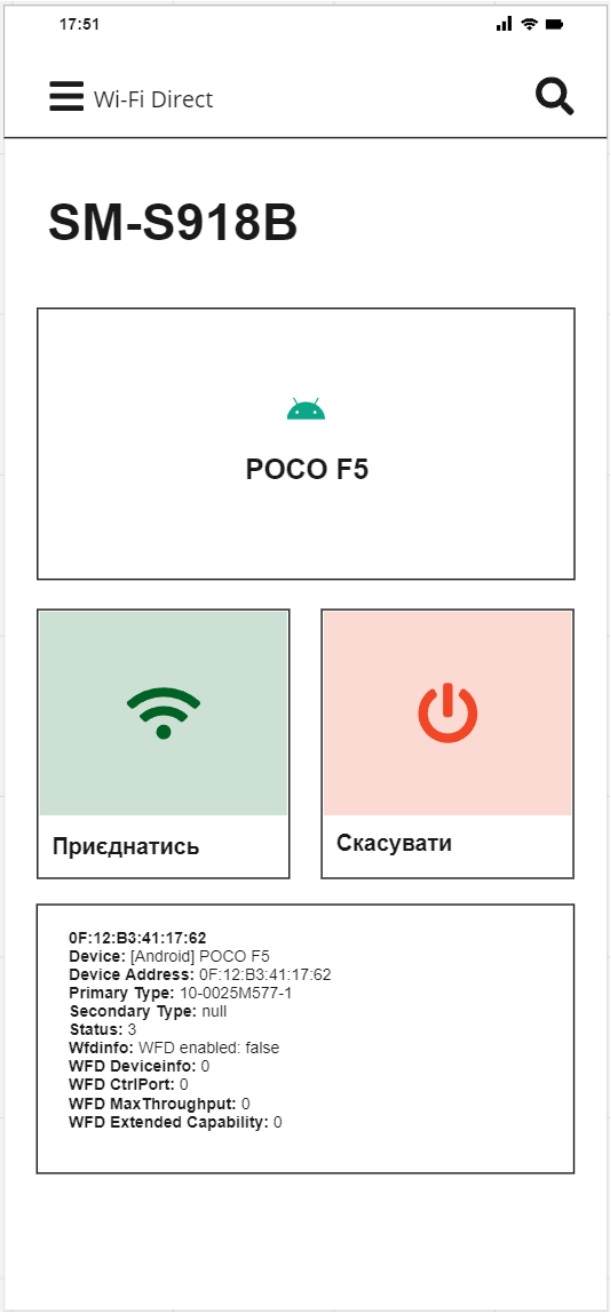

Рис. 4.4. Інформація про пристрій

- Device Address - MAC-адреса пристрою;
- Primary type - тип пристрою (може бути телефон, принтер та інше);
- Secondary type - додатковий атрибут до primary type;
- Status - статус підключення пристрою;
- WFD Info - підключення активне чи ні;
- WFD DeviceInfo - інформація про пристрій;
- WFD CtrlPort - порт;

- WFD MaxThroughput - максимальна пропускна здатність;
- WFD Extended Capability - додаткові можливості для зображень.

Натиснувши на кнопку "Приєднатись" наші пристрої зможуть приєднатися один до одного.

Коли підключення відбудеться успішно, статус пристрою зміниться на "Приєднано" і з'явиться можливість передавати інформацію. Результат показано на рис. 4.5.

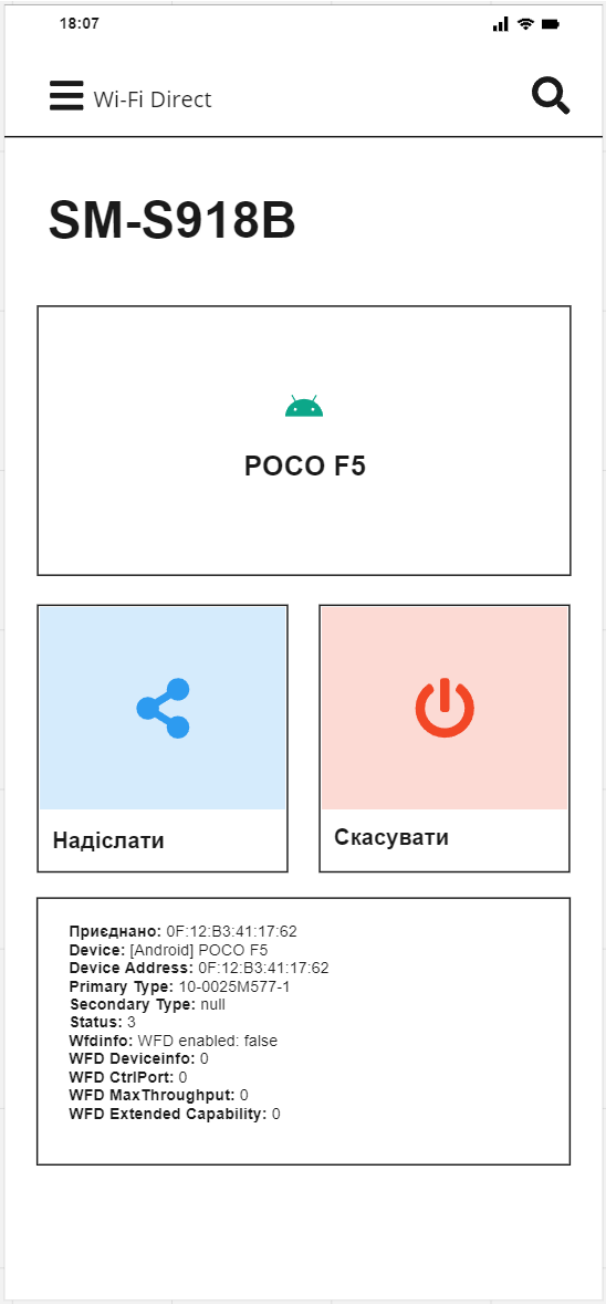

Рис. 4.5. Пристрої під'єднані

Далі можна надсилати файли, натиснувши кнопку "Надіслати", після чого відкриється пам'ять телефону і можна передати файл.

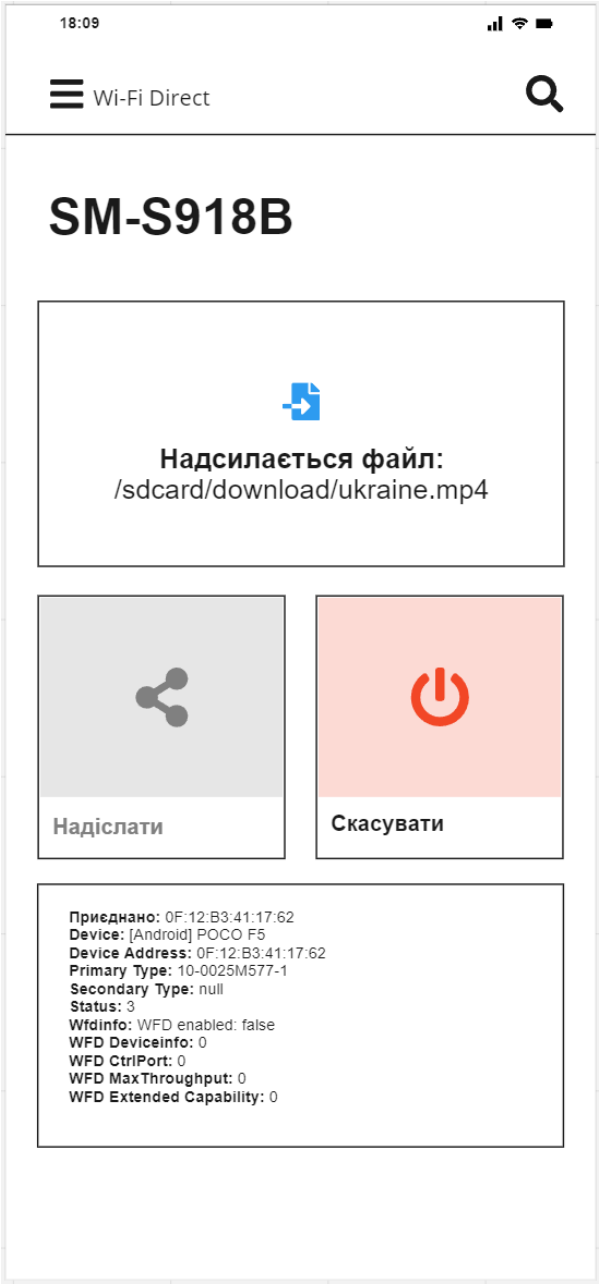

Рис. 4.6. Процес надсилання файлу

На рис. 4.6 показано процес надсилання файлу.

Унаслідок тестування було продемонстровано можливості пошуку сусідніх вузлів, отримання докладної інформації про них, підключення до вузлів і передавання даних конкретному вузлу. Так само було продемонстровано користувацький інтерфейс. Як макети були використані стандартні фрейми Android Studio.

РОЗДІЛ 5
ОХОРОНА ПРАЦІ

У цій частині дипломної роботи суб'єктом є інженер-дослідник, який займається розробкою моделей мереж і мобільних додатків.

Інженер-дослідник може зіткнутися з потенційно небезпечними фізичними та психологічними проблемами під час виконання своїх обов'язків, оскільки робота, пов'язана з обробкою та аналізом досліджуваних даних, здійснюється за допомогою персонального комп'ютера.

Він вважає, що, оскільки небезпечні та шкідливі виробничі фактори присутні на робочому місці, необхідно створити заходи з охорони праці, щоб створити безпечні та нешкідливі робочі місця.

На другому поверсі знаходиться відділ планування мережі радіодоступу, де знаходиться робоче місце інженера-дослідника.

5.1. Аналіз небезпечних і шкідливих факторів, що впливають на інженера-дослідника

Організація робочого місця програміста. Приміщення програміста завдовжки 8 метрів, завширшки 5 метрів, має загальну площу 40 метрів квадратних і висоту стелі 3,5 метрів. У приміщенні є шість робочих місць з комп'ютерами. Кожне робоче місце має робочий стол площею 1.44 м^2, стілець і персональний комп'ютер із монітором, системним блоком, клавіатурою та мишею. Згідно з [39], одне робоче місце має мати площу не менше 6,0 кв. м і об'єм не менше 20,0 куб. м. Отже, цього приміщення достатньо для розміщення шести робочих місць операторів ПК.

Термін «небезпечний» використовується для опису факторів, які негайно погіршують здоров'я працівника. Чинники, які безпосередньо або побічно призводять до порушення працездатності або здоров'я працівників, називають шкідливими факторами виробничого середовища.

На програміста впливають наступні потенційно небезпечні та шкідливі фактори [40]: мікроклімат, недостатнє освітлення на робочому місці та статична електрика.

Мікроклімат місця роботи програміста Робота програміста належить до категорії Іа або Іб, тому вона повинна відповідати наступним вимогам [41]:

Таблиця 5.1

Оптимальні величини температури, відносної вологості та швидкості руху повітря в робочій зоні виробничих приміщень

Період Року	**Категорія Робіт**	**Температура повітря**	**Відносна вологість**	**Швидкість руху, м/сек.**
Холодний період року	Легка Іа	22 - 24	60 - 40	0,1
	Легка Іб	21 - 23	60 – 40	0,1
Теплий період року	Легка Іа	23 - 25	60 - 40	0,1
	Легка Іб	22 - 24	60 - 40	0,2

За допомогою пристрою, відомого як психрометр Августа, температура та вологість у відділі інформаційних технологій відповідають показникам для теплого періоду року. Розташовані в приміщенні 6 ПК є джерелами тепловиділення, а нагріті поверхні опалювальної системи використовуються для створення ідеального мікроклімату в приміщенні в холодний період року. Гранично допустима густина потоку енергії $I_{г.д}$, Вт/м^2, яка визначається на основі площі опромінюваної поверхні тіла людини ($S_{орг}$), є нормативним показником ІЧВ. $I_{г.д.}$ має бути 35 Вт/м^2 при $S_{опр} > 50\%$; Іг.д. має бути 70 Вт/м2 при $S_{опр} \sim 25\text{-}50\%$; і $I_{г.д.}$ має бути 100 Вт/м^2 при $S_{опр} < 25\%$.

Природне та штучне освітлення. Коефіцієнт природного освітлення (КПО) є параметром природного освітлення, як було визначено в [42]. КПО визначаються за типом зорових робіт, які виконуються. Роботи програміста відносяться до середньої точності (IV розряд зорових робіт із мінімальним розміром об’єкта

розрізнення 0,5–1,0 мм) з КПО=1,5% при використанні бокового освітлення. Для штучного освітлення стандартними параметрами є Емін, мінімальний рівень освітленості, і КП, коефіцієнт пульсації світлового потоку, який не повинен перевищувати 20%. Розділ зорових робіт визначає мінімальну освітленість. Відповідно до IV розряду зорових робіт вона становить 300-500 лк.

Виробничі випромінювання. Таблиця 5.2 показує допустимі значення параметрів неіонізуючих електромагнітних випромінювань від монітора комп'ютера. Потужність експозиційної дози є нормованим параметром невикористаного рентгенівського випромінювання. Рівень чутливості монітора не повинен перевищувати 100 мкР/год на відстані 5 см від його поверхні. Зазвичай на робочому місці програміста рівень рентгенівського випромінювання не перевищує 20 мкР/год.

Таблиця 5.2

Допустимі значення параметрів неіонізуючих електромагнітних випромінювань

Найменування параметра	**Допустимі значення**
Напруженість електричної складової електромагнітного поля на відстані 50 см від поверхні монітора ПК	10 Вт/м
Напруженість магнітної складової електромагнітного поля на відстані 50 см від поверхні монітора ПК	0.3 А/м
Напруженість для операторів ПК не повинна перевищувати	20 кВ/м

Напруженість електричної складової може досягати 6 В/м на відстані 5-10 см від екрана та корпуса монітора, що не перевищує допустимі значення.

Електробезпека. Статична електрика. Приміщення ІТ-відділу можна віднести до 1 класу небезпеки ураження електричним струмом, якщо воно не є надзвичайно небезпечним (сухе, без пилу, нормальна температура повітря, ізольовані підлоги та мало заземлених приладів).

На робочому місці програміста є лише стандартний корпус системного блоку комп'ютера IBM. Для знешкодження статичної електрики на системному блоці повинно бути встановлене заземлення, згідно з [43]: пункт 5 «Заходи по захисту від статичної електрики». Системний блок не відповідає вищевказаним нормам, оскільки на ньому відсутнє заземлення.

Основні причини ураження людей електричним струмом на робочому місці включають: дотик до металевих неструмоведучих частин, таких як корпус комп'ютера, які можуть бути під напругою через ушкодження ізоляції; нерегламентоване використання електричних приладів; і відсутність інструктажу з правил електробезпеки для співробітників.

5.2. Організаційно-технологічні та конструктивні заходи для зменшення впливу шкідливих виробничих факторів

Нормалізація повітря робочої зони. Для створення та автоматичної підтримки в ІТ-відділі необхідно дотримуватися ідеальних температур, вологості, чистоти та швидкості руху повітря незалежно від зовнішніх умов. У холодні роки використовується водяне опалення, а в теплі роки використовується кондиціонування повітря [44].

Виробниче освітлення. У результаті аналізу освітлення на робочому місті програміста було виявлено, що воно не відповідає встановленим нормам. Тому, щоб покращити робочі умови, ми рекомендуємо встановити п'ять додаткових світильників і збільшити загальну кількість ламп, щоб досягти вищезазначеного значення, тобто 36 світлодіодних ламп. Крім того, необхідно планувати очищення віконних блоків і світильників не менше двох разів на рік, щоб забезпечити чисте освітлення [42].

Електробезпека. Пропоную забезпечити електробезпечність у приміщенні ІТ-відділу за допомогою наступних технічних методів і засобів захисту: зволожувачі та нейтралізатори, антистатичне покриття підлоги; забезпечити заземлення

металевих корпусів устаткування. Заземлення корпусу комп'ютера дозволяє підвести жилу, що заземлює, до розеток. Для електроустановок з напругою до 1000 В необхідний опір заземлення 4 Ом. Крім того, необхідні організаційні заходи, такі як швидке проведення інструктажів з техніки безпеки [45].

Ергономіка та організація робочого місця. Після оцінки робочого місця програміста в ІТ-відділі було встановлено, що воно відповідає вимогам.

Виходячи з результатів аналізу важкості та напруженості праці, пропоную скоротити час роботи за комп'ютером до п'ятдесяти хвилин протягом восьми годин робочого дня [46].

5.2.1. Розрахунок освітленості робочого місця інженера-дослідника на відповідність розряду зорової роботи

За даними вимірювань (люксметр Ю-116), рівень природної освітленості поверхні, де розташований комп'ютер програміста, складає 200 лк, тоді як освітленість цієї ж поверхні відкритим небосхилом складає 20000 лк. Отже, КПО = 1%, що не відповідає нормативному КПО.

Світлодіодні лампи T8 G13 використовуються для штучного освітлення у приміщенні. Вони мають багато переваг порівняно з люмінесцентними та лампами розжарювання, включаючи більшу світлову віддачу (у 2-5 разів більшу, ніж у ламп розжарювання) і триваліший термін служби (до 10 000 годин) [42].

Проведемо розрахунок штучного освітлення для кімнати площею 40 квадратних метрів, яка має ширину 5 метрів, довжину 8 метрів і висоту 3,5 метрів, використовуючи метод коефіцієнта використання світлового потоку.

Для визначення потрібної кількості світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою:

$$F=E*S*K*Z/n \quad (5.1)$$

(де **F** – світловий потік, що розраховується, Лм; **E** – нормована мінімальна освітленість, Лк; E = 300 Лк; **S** – площа освітлюваного приміщення (у нашому випадку

S=40 м^2); **Z** – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1,1...1,2, в нашому випадку Z=1,1); **K** – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників в процесі експлуатації (його значення залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку К=1,5); **n** – коефіцієнт використання світлового потоку, (виражається відношенням світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп, і обчислюється в долях одиниці;) залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ($\rho_{ст.}$) і стелі ($\rho_{стелі}$), значення коефіцієнтів дорівнюють $\rho_{ст}$ = 40% і $\rho_{стелі}$=60%.

Обчислимо індекс приміщення за формулою:

$$i=S/h(A+B) \qquad (5.2)$$

(де **S** – площа приміщення, S=40 м^2; **h** – розрахункова висота підвісу, h = 3.3 м; **A** – ширина приміщення, A = 5 м; **B** – довжина приміщення, B = 8 м.)

Підставивши значення отримаємо: **i=40/3.3(5+8)=0.93**. Знаючи індекс приміщення, знаходимо **n=0.22**. Підставимо всі значення у формулу для визначення світлового потоку **F**:

F=(300*1.5*40*1.1)/0.22=90000 Лм.

Для освітлення використані світлодіодні лампи з матовим покриттям типу LRC-T8-S1500G13-220-22,0W, світловий потік яких **$F_л$ = 2500** Лм.

$$N=F/F_л \qquad (4.3)$$

(де **N** – визначуване число ламп; **F** – світловий потік, **F=90000 Лм**; **Fл** – світловий потік однієї лампи, **$F_л$ = 2500 Лм**.)

N=90000/2500=36

В приміщенні використовуються світильники типу ЛПО. Кожен світильник комплектується чотирма лампами. Тобто необхідно використовувати 9 світильників із 36 працюючими лампами в них.

У ІТ відділі авіапідприємства, де аналізувалось робоче місце програміста працює 5 світильників з 20 лампами в них, тому рівень штучного освітлення не задовольняє санітарним нормам.

5.3. Пожежна безпека

Згідно з [47], приміщення ІТ-відділу центрального офісу авіапідприємства відноситься до категорії Д «Негорючі речовини та матеріали в холодному стані». місця, де знаходяться ГР систем машин, охолодження та гідроприводу, устаткування з масою не більше 60 кг при тиску не більше 0,2 мПа, кабелі електропроводки до устаткування, окремі меблі на місцях

Центральний офіс ІТ-відділу по пожежній небезпеці будівельних конструкцій відноситься до категорії К1 (малопожежонебезпечні), оскільки тут знаходяться займисті речовини (книги, документи, меблі, оргтехніка тощо) і важкогорючі речовини (сейфи, різне устаткування тощо), які можуть горіти без вибуху, якщо вони стикаються з вогнем.

Будинки можна класифікувати за конструктивними характеристиками як будинки з несучими та огороджуючими конструкціями, виготовленими з природних або штучних каменів, бетону або залізобетону; для перекриттів допускається використання дерев'яних конструкцій, захищених штукатуркою або важкогорючими листовими, а також плитних матеріалів.

Таким чином, будинок Центрального офісу має третю ступінь вогнестійкості (III).

Приміщення ІТ-відділу авіапідприємства має клас пожежної небезпеки Ф 4.2.

Причини виникнення пожежі. Пожежі в ІТ-відділі можуть спричинити жахливі наслідки, такі як смерть людей, втрата цінної інформації та шкода майну. Тому необхідно: виявити та усунути всі фактори, які можуть спричинити пожежу; розробити план ліквідації пожежі; і розробити план евакуації людей.

Причинами виникнення пожежі можуть бути:

– несправності електропроводки, розеток і вимикачів які можуть привести до короткого замикання або пробою ізоляції;

– використання ушкоджених (несправних) електроприладів;

– використання в приміщенні електронагрівальних приладів з відкритими нагрівальними елементами;

– виникнення пожежі внаслідок влучення блискавки в будинок;

– загоряння будинку внаслідок зовнішніх впливів;

– неакуратне поводження з вогнем і недотримання мір пожежної безпеки.

Засоби пожежогасіння та пожежно-охоронної сигналізації. Як сказано в [47]: «3.3. На кожному підприємстві, з урахуванням його пожежної небезпеки, повинен бути встановлений відповідний протипожежний режим. Цей режим повинен включати: організацію експлуатації та обслуговування наявних технічних засобів протипожежного захисту (протипожежного водопроводу, насосних станцій, пожежних сигналізацій, автоматичного пожежогасіння, димовидалення, вогнегасників тощо). В приміщенні встановлено один переносний вуглекислотний вогнегасник ВВК-5, якого достатньо для цього типу приміщення та площі. Крім того, на стелі встановлено два бездротові ІЧ-датчики диму Страж М-501, розраховані на площу 40 квадратних метрів.

Якщо виникне пожежа, спрацює протипожежна сигналізація, необхідно відключити електроживлення, викликати пожежну команду за номером 101, вивести людей із приміщення відповідно до плану евакуації, представленого на рисунку 5.1, і використовувати вогнегасники для ліквідації пожежі. При наявності невеликого вогнища можна використовувати підручні засоби, щоб заблокувати доступ повітря до вогнища.

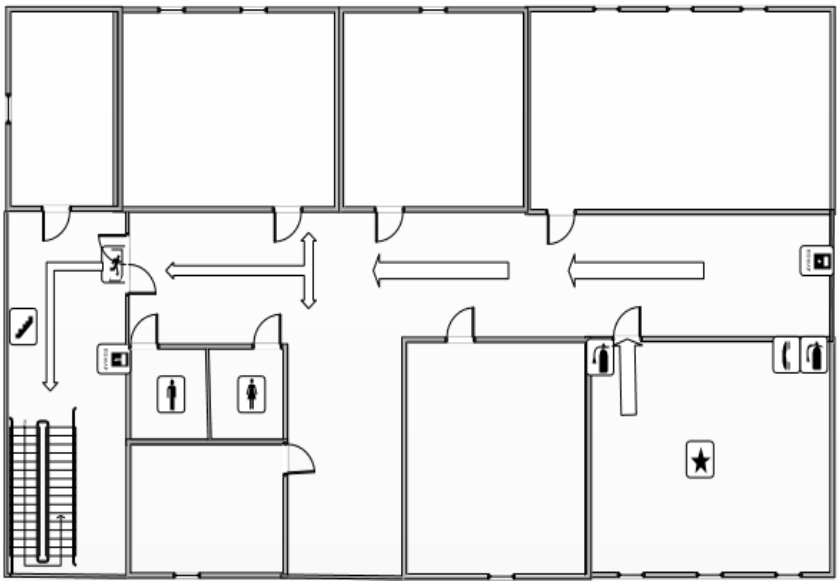

Рис. 4.1 План евакуації з приміщення ІТ відділу компанії

5.4. Інструкція з охорони праці при роботі з персональним комп'ютером

Вимоги безпеки перед початком роботи.

- Перед початком роботи працівник повинен зовнішнім оглядом перевірити цілісність корпусів системного блоку, відео монітора, принтера, клавіатури.
- Перевірити цілісність кабелів живлення, місць їх підключення (розеток електромережі, продовжувачів електромережі, розгалужувальних коробок, штепсельних вилок).
- Підготувати своє робоче місце, прибравши речі, які можуть заважати при виконанні роботи.
- Ввімкнути живлення ПК.
- У випадку, якщо після ввімкнення ПК не проходить загрузка або комп'ютер не виходить на робочий режим, працівник повинен повідомити керівника чи спеціаліста відділу інформаційних технологій.
- При виявленні ушкодження або яких-небудь інших недоліків повідомити безпосереднього керівника. Не приступати до роботи без його вказівки.

Вимоги безпеки під час роботи

- Усі компоненти столу, включаючи клавіатуру, повинні бути стійко розташовані. Крім того, необхідно передбачити можливість переміщення клавіатури. Її розміщення

та кут нахилу повинні відповідати потребам користувача ПК. Якщо в конструкції клавіатури немає місця для опору долонь, клавіатура повинна бути розташована на відстані не менше 100 мм від краю столу, щоб забезпечити оптимальну відстань для моніторного поля. Коли ви працюєте на клавіатурі, вам слід сидіти прямо, не напружуючись.

- Для зменшення несприятливого впливу на користувача пристроїв типу "миша" (вимушена поза, необхідність постійного контролю за якістю дій) слід забезпечити вільною більшу площу поверхні столу для переміщення "миші" і зручного упору ліктьового суглоба.
- Не припустимі сторонні розмови, роздратовуючи шуми тощо.
- Періодично при вимкнутому ПК слід видаляти злегка зволоженою мильним розчином хлопко-паперовою салфеткою пил з поверхонь апаратури. Екран і захисний екран протирають ватою, зволоженою спиртом.
- Не дозволяється використовувати рідинні або аерозольні засоби чистки поверхонь ПК.

Забороняється:

- самостійно ремонтувати апаратуру, в яких кінескоп та інші елементи можуть знаходитись під високою напругою (до 25 кВ0.)
- класти будь-які речі на апаратуру ПК, бутерброди та напої на клавіатуру або поруч з нею. Це може вивести її з ладу;
- затуляти вентиляційні отвори в апаратурі, це може призвести до її перегріву і виходу з ладу.

- Для того, щоб зменшити негативний вплив факторів ризику, пов’язаних з роботою на ПК, на стан здоров’я працівників, передбачаються додаткові регламентовані перерви для відпочинку користувачів ПК:
  - через кожний час безперервної роботи – 10 хвилин;
  - через кожні 2 години – 15 хвилин.
- При можливості слід чергувати зміну діяльності з іншою, не пов’язаною з роботою на ПК.

- З метою зменшення негативного впливу монотонності доцільно застосовувати чергування операцій введення тексту і введення даних (зміна змісту і темпу роботи) і т.п.
- При роботі на лазерних принтерах:
- Розташовувати принтер необхідно поряд з системним блоком так, щоб з'єднувальні шнури не були натягнуті. Забороняється ставити принтер на системний блок.
- Перш, ніж програмувати роботу принтера, впевніться, що він знаходиться в режимі зв'язку з системним блоком.
- Для досягнення високоякісного, чистого, з високою роздільною здатністю зображення щоб не зіпсувати апарат, потрібно використовувати папір, марка якого вказана в інструкції до принтера (найчастіше папір вагою 60-135 г/м^2, типу Canon або Xerox 4024).
- Обрізання країв паперу повинно бути виконаним гострим лезом ножа, без зазусенців – це зменшить вірогідність загинання паперу.
- При виконанні роботи (більше 20 хвилин), коли втручання користувача в роботу програми не потрібне, бажано вимикати живлення відео монітора.
- Для підтримки загального тонусу м'язів, профілактики кістково-м'язових порушень, зорового дискомфорту та інших несприятливих суб'єктивних почуттів під час регламентованих перерв необхідно виконувати комплекси рекомендованих вправ для очей, для хребта, для рук.
- Кількість мікро пауз до 1-2 хвилин слід визначити індивідуально. Форма та зміст перерв можуть бути різними виконання допоміжних робіт, не пов'язаних з роботою ПК , приймання їжі, виконання рекомендованих вправ.
- Виконання фізичних вправ протягом дня рекомендується індивідуально, залежно від почуття втоми. Гімнастика повинна біти на корекцію вимушеної пози покращення кровообігу, часткову компенсацію, дефіциту рухової активності.
- Про виявлені несправності (іскріння, пробоїв, запаху гару, ознак горіння тощо) негайно припинити роботу, відключити все обладнання від електромережі і терміново повідомити безпосереднього керівника або спеціаліста по ремонту ПК.

Вимоги безпеки при закінчення роботи на ПК.

- Закінчити і зберегти в пам’яті ПК файли, які знаходились у роботі. Виконати всі дії для коректного завершення роботи в оперативній системі.
- Вимкнути принтер та інші периферійні пристрої, вимкнути системний блок. При наявності пристрою безперебійного живлення (ПБЖ) вимкнути його живлення.
- Вимкнути ПК кнопкою «POWER” (ЖИВЛЕННЯ) та вийняти штепсельну вилку кабелю живлення з розетки
- Накрити клавіатуру кришкою для попередження попадання в неї пилу.
- Навести порядок на робочому місці.

Висновки

У цьому розділі розглядаються способи, за допомогою яких небезпечні та шкідливі виробничі фактори можуть впливати на технічний персонал відділу ІТ. Розраховано освітленість робочої зони. Ми отримали дев’ять світильників із 36 світлодіодними лампами, які є ідеальним варіантом для освітленості робочої зони та не порушують стандарти освітленості 300-500 лк. В ІТ-відділі були представлені інструкції з охорони праці при роботі з персональними комп’ютерами та рекомендації щодо пожежної безпеки.

РОЗДІЛ 6
ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

5.1. Аналіз впливу техногенних чинників на навколишнє природне середовище

У результаті діяльності людини в середовищі існування воно поступово змінювало свій вигляд, що призвело до руйнування біосфери та створення штучного середовища, відомого як техносфера. Науковці стверджують, що сьогодні майже все середовище, в якому живе людина, є техногенним. Техносфера, яку людина створила, охоплює майже всю планету і навіть вийшла за межі Землі.

Техногенне середовище, також відоме як техносфера, складається з наслідків діяльності людини.

Люди постійно виконують два основні завдання, коли вони працюють у техногенному середовищі: створювати та використовувати системи, які захищають їх від шкідливих факторів.

Негативні фактори навколишнього середовища впливають на людину як прямо, так і непрямо [48].

5.2. Принципи роботи базових станцій і стільникових пристроїв, а також їх негативний вплив на навколишнє середовище

У сучасному світі практично неможливо уникнути впливу ЕМП через стрімке зростання технологій і приладів.

ЕМП стільникового зв’язку походить від телефонних трубок і базових станцій супроводу стільникового зв’язку. Ці джерела ЕМП працюють по-різному на кожного. Стільниковий телефон як джерело ЕМП має можливість максимально наближатися до голови користувача на відстань від двох до п’яти сантиметрів у неконтрольованих

умовах. Головний мозок, периферичні рецепторні зони вестибулярного апарату, слухові аналізатори та сітківка очей піддаються впливу ЕМП. Крім того, люди, з якими споживач розмовляє по телефону, також піддаються впливу випромінювання стільникового телефону.

Електромагнітні поля створюються на базових станціях імпульсно. Це залежить від часу доби, насиченості покриття базових станцій і кількості базових станцій, розташованих у певній зоні. Саме базові станції генерують техногенне електромагнітне поле, яке покриває всю область дії стільникового зв'язку. Базові станції мають низькоінтенсивне електромагнітне поле радіочастотного діапазону, яке впливає на людей щодня.

За даними екологів і лікарів-гігієністів, добре відомо, що будь-який діапазон електромагнітного випромінювання має серйозні наслідки для здоров'я людини та її здатності працювати. Поширеність електромагнітних полів робить їх більш небезпечними для людини, ніж радіація. Електропроводка, освітлення, побутові електроприлади, лінії електропередач тощо створюють електричні поля промислової частоти, які оточують людей щодня.

У зв'язку зі стрімким розширенням мережі джерел фізичних полів електромагнітної природи та збільшенням їх потужностей енергетичне навантаження від електромагнітних випромінювань як у промисловості, так і в побуті постійно зростає. Хоча людина фізично не може відчувати електромагнітне поле, воно знижує адаптивні резерви, імунітет, працездатність, викликає синдром хронічної втоми та підвищує ризик захворювань. Діти, підлітки, вагітні жінки та люди з ослабленим здоров'ям особливо уразливі до електромагнітних випромінювань.

Вплив електромагнітного поля на клітину. Електромагнітне поле впливає на заряджені частинки і струми, внаслідок чого енергія поля на рівні клітини перетворюється в інші види енергії.

Цітогенетичні (вихід хромосомних аберацій) дослідження показали, що клітини з порушеннями в експериментальній групі були більші, ніж у контрольній групі. Опромінення ЕМП повітряно-сухого насіння і проростків салату також

призвело до збільшення хромосомних аберацій. Цитогенетичні дослідження клітин крові корів з ферми показали, що є більше генетичних ушкоджень і аномальних гематопоезу [49].

Вплив електромагнітного поля на тканини. На зміни в живій тканині також впливають слабкі електромагнітні поля з меншим порогом теплового ефекту. Були проведені дослідження щодо впливу стільникового телефону, комп'ютерного блоку та інших електронних пристроїв на організм людини. В результаті цих досліджень було виявлено, що вплив цих джерел проявляється в тому, що регенерація тканин погіршується.

В електричному полі атоми та молекули поляризуються, а полярні молекули орієнтуються в напрямку розповсюдження магнітного поля. У результаті змінної поляризації діелектрика (сухожиль, хрящів, кісток) і струмів провідності змінне електричне поле нагріває тканини живих організмів.

Вплив електромагнітного поля на нервову систему. В радянські часи було проведено перше експериментальне дослідження того, як електромагнітне поле впливає на нервову систему. Електромагнітне поле впливає на мембрани нейронів, пам'ять, умовно-рефлекторну діяльність і мозок. У моделях показано, як слабкі електромагнітні поля можуть впливати на процеси синтезу в нервових клітинах. Коркові нейрони демонструють значні зміни імпульсації, що призводить до порушення переданої інформації в більш складні структури мозку. Дії електромагнітного поля в надвисокочастотному діапазоні можуть спричинити порушення короткочасної пам'яті.

Вплив електромагнітного випромінювання на імунну систему. Наразі є достатньо інформації про те, що електромагнітне поле порушує процеси імуногенезу. Електромагнітне поле змінює характер інфекційного процесу, порушує білковий обмін, зменшує вміст альбумінів і збільшує гамма-глобуліни в крові. Крім того, електромагнітне поле може виступати як пусковий фактор або алерген, що викликає сильні реакції у хворих алергіків.

Вплив електромагнітного поля на статеву систему. Під впливом електромагнітного випромінювання знижується функція сперматогенезу, змінюється менструальний цикл, уповільнюється ембріональний розвиток, виникають вроджені каліцтва у новонароджених дітей і зменшення лактації у годуючих матерів.

Вплив електромагнітного поля на рослини. Численні дослідження показали, що електромагнітні хвилі мають значний вплив на біологічні об'єкти через низку індукованих ефектів. Як слабкі, так і сильні ЕМП впливають на морфологію, фізіологію, біохімію та біофізичні характеристики рослин. Вони впливають на зростання, розвиток і розмноження рослин.

Рівень електричного поля, реєстрованого поблизу повітряних ліній (ПЛ), теоретично достатній для пошкодження листя рослин. Експерименти та спостереження, проведені щодо впливу ЕМП ліній електропередачі на рослини, показали, що в порівнянні з контролем суха маса надземної маси рослин вівса та соняшника, зростаючих під ПЛ, була нижчою. ЕМП має негативний вплив на потенційну нітрогенезну активність різосферної популяції та довжину проростків рослин.

Вплив слабких електромагнітних полів на живі організми. На зміни в живій тканині також впливають слабкі електромагнітні поля з меншим порогом теплового ефекту. Ряд наукових установ проводить дослідження біологічного впливу стільникових телефонів, комп'ютерних блоків та інших електронних пристроїв. У цьому процесі перевірялися небезпеки електронних пристроїв як у робочому, так і у вимкненому стані, а також без джерела живлення [50].

Результати досліджень, які вивчали вплив стільникового телефону, комп'ютера та інших сучасних радіоелектронних пристроїв на різні організми як у робочому, так і у вимкненому стані, виявилися невтішними. Ці дослідження показали, що ці пристрої мають надзвичайно негативний вплив на стан біологічних об'єктів:

- у зниженні рухової активності і виживання мікроорганізмів;
- у збільшенні смертності мікроорганізмів;
- в погіршенні регенерації тканин;
- в порушенні ембріонального і личинкового розвитку;
- у зниженні біохімічних реакцій, порушення метаболізму;

- у зниженні енергетичного потенціалу у всіх життєво важливих системах організму.

5.3. Методи та засоби захисту навколишнього середовища від впливу техногенних чинників

Захист від електромагнітних випромінювань. Потрібно вжити низку заходів, щоб захистити людей і населення в зоні дії радіоелектронних засобів від впливу ЕМВ. Це можуть бути організаційні, інженерно-технічні та лікарсько-профілактичні дії.

Органи санітарного нагляду відповідають за організацію та інженерно-технічну роботу. Санітарні лабораторії підприємств і установ, які використовують джерела електромагнітного випромінювання, повинні брати участь у гігієнічних оцінках нового будівництва та реконструкції об'єктів, які виробляють і використовують радіозасоби, а також нових технологічних процесів і обладнання з використанням ЕМП. Вони також повинні проводити поточний санітарний нагляд за об'єктами, які використовують джерела електромагнітного випромінювання, виконувати організаційно-методич

Необхідно забезпечити, щоб опромінюючі та опромінюючі об'єкти розташовувалися таким чином, щоб зменшити інтенсивність опромінення, ще на етапі проєктування. У зв'язку з тим, що повністю уникнути опромінення неможливо, необхідно зменшити ймовірність проникнення людей у місця з високою інтенсивністю ЕМП, щоб зменшити час, який люди проводять під опроміненням.

Колективні (група будинків, район, населений пункт) і локальні (окремі будівлі, приміщення) і інженерно-технічні методи та засоби захисту мають вирішальне значення. Розрахунок поширення радіохвиль у конкретній місцевості є основою колективного захисту. Природні екрани, такі як складки місцевості, лісонасадження та нежитлові будівлі, є найбільш економічно ефективними.

Встановлення антени на горі може значно зменшити інтенсивність поля, яке опромінює населений пункт. Наприклад, для високоспрямованих антен, орієнтація діа-

грами спрямованості змінюється відповідно, збільшуючи висоту антени. Але висока антена менш стійка, складніша та дорожча. Крім того, цей захист погано працює з відстані.

Затухання хвилі, яка проходить через екран (наприклад, через лісову смугу), має бути враховано при захисті екрана від випромінювання. Рослинність може служити екраном. Відбивні та радіопоглинальні щити, які використовуються як спеціальні екрани, дорогі та неефективні.

Локальний захист використовується часто, оскільки він дуже ефективний. Використання радіозахисних матеріалів забезпечує високе поглинання енергії випромінювання у матеріалі та віддзеркалення його поверхні. Екранування шляхом віддзеркалення використовується за допомогою металевих листів і сіток з хорошою провідністю. Металізовані стіни можна обклеювати металізованими шпалерами, віковими сітками та шторами [51].

Хоча в цьому місці немає багато випромінювання, віддзеркалене екранами випромінювання поширюється по простору та потрапляє на інші предмети.

Персонал, який працює на невеликій відстані, повинен бути захищений шляхом екранування апаратури.

Поруч із віддзеркалюючими є екрани, виготовлені з матеріалів, що поглинають випромінювання. Існує широкий спектр радіопоглинальних матеріалів, як однорідних, так і композиційних, які складаються з різноманітних діелектричних і магнітних речовин. Поглинальна поверхня екрана виготовляється шорсткою, ребристою або у вигляді шипів, щоб підвищити продуктивність. Радіопоглинаючі матеріали можуть захистити навколишнє середовище від ЕМП, який виробляє джерело в екранованому об’єкті.

Засоби індивідуального захисту використовуються лише тоді, коли інші засоби захисту недоступні або недостатньо ефективні: під час переходу через зони збільшеної інтенсивності випромінювання, під час ремонту та налагодження в аварійних ситуаціях, під час короткочасного контролю та при зміні інтенсивності опромінення. Такі засоби незручні в експлуатації, обмежують можливості виконання операцій і погіршують умови гігієни.

Для захисту тіла використовується одяг із металізованих тканин та радіопоглинаючих матеріалів. Металізована тканина складається з бавовняних чи капронових ниток, спіральне обвитих металевим дротом, таким чином, ця тканина, мов металева сітка послаблює випромінювання не менш, як на 20-30 дБ. При зшиванні деталей захисного одягу потрібно забезпечити контакти ізольованих провідників. Тому електрогерметизація швів проводиться електропровідними розчинами чи клеями, які забезпечують гальванічний контакт або збільшують ємнісний зв'язок проводів, котрі не контактують.

Очі захищають спеціальними окулярами зі скла з нанесеною на внутрішній бік провідною плівкою двоокису олова. Гумова оправа окулярів має запресовану металеву сітку або обклеєна металізованою тканиною; цими окулярами випромінювання НВЧ послаблюється на 20-30 дБ.

Раніше використовували рукавички та бахіли, проте зараз вважають непотрібними, оскільки допустима величина щільності потоку енергії для рук та ніг у багато разів вища, ніж для тіла.

Колективні та індивідуальні засоби захисту можуть забезпечити тривалу безпечну роботу персоналу на радіооб'єктах [51].

Колективний та індивідуальний захист від шуму. Боротьба з шумом в джерелі його виникнення. Це найбільш дієвий спосіб боротьби з шумом. Створюються малошумні механічні передачі, розроблено способи зниження шуму в підшипникових вузлах, вентиляторах.

Зниження шуму звукопоглинанням. Об'єкт, котрий випромінює шум, розташовують у кожусі, внутрішні стінки якого покриваються звукопоглинальним матеріалом. Кожух повинен мати достатню звукопоглинальну здатність, не заважати обслуговуванню обладнання під час роботи, не псувати інтер'єр цеху. Різновидом цього методу є кабіна, в котрій розташовується найбільш шумний об'єкт і в котрій працює робітник. Кабіна зсередини вкрита звукопоглинальним матеріалом, щоб зменшити рівень шуму всередині кабіни, а не лише ізолювати джерело шуму від решти виробничого приміщення.

Зниження шуму звукоізоляцією. Суть цього методу полягає в тому, що шумовипромінювальний об'єкт або декілька найбільш шумних об'єктів розташовуються окремо, ізо-

льовано від основного, менш шумного приміщення звукоізолювальною стіною або перегородкою. Звукоізоляція також досягається шляхом розташування найбільш шумного об'єкта в окремій кабіні. При цьому в ізольованому приміщенні і в кабіні рівень шуму не зменшиться, але шум впливатиме на менше число людей. Звукоізоляція досягається також шляхом розташування оператора в спеціальній кабіні, звідки він спостерігає та керує технологічним процесом. Звукоізоляційний ефект забезпечується також встановленням екранів та ковпаків. Вони захищають робоче місце і людину від безпосереднього впливу прямого звуку, однак не знижують шум в приміщенні.

Зниження шуму акустичною обробкою приміщення. Акустична обробка приміщення передбачає використання звукопоглинальних матеріалів для покриття стелі та верхніх стін. Як наслідок цього, інтенсивність відбитих звукових хвиль зменшується. Стелі можуть мати звукопоглинальні щити, конуси, куби або резонаторні екрани (штучні поглиначі). Застосовувані матеріали та конструкції, місце розташування, розмір, геометрія та розташування джерел шуму впливають на ефективність акустичної обробки приміщень. Витягнута форма має більший ефект у низьких приміщеннях, де висота стелі не перевищує 6 м. Зниження шуму на 8 дБА можна досягти за допомогою акустики [52].

У процесі проєктування промислових об’єктів і обладнання необхідно враховувати заходи щодо зниження шуму. Особливу увагу слід приділити переміщенню шумного обладнання в окреме приміщення, щоб зменшити кількість працівників, які працюють у приміщеннях з високим рівнем шуму, а також здійсненню заходів, спрямованих на зниження шуму при мінімальних витратах коштів, обладнання та матеріалів. Зниження шуму залежить від знешумлення всього обладнання з високим рівнем шуму.

Створення шумових карт і спектрів шуму обладнання та виробничих приміщень є першим кроком до знешумлення робочого обладнання в приміщенні. За допомогою цих карт визначається напрямок роботи.

Висновок

Розвиток електроніки та радіотехніки призвів до забруднення природного середовища електромагнітними випромінюваннями, також відомими як поля. Радио-, телевізійні

та радіолокаційні станції є їхніми основними джерелами. Телевізійні центри або ретранслятори, радіоцентри та засоби радіозв'язку різного призначення розташовані поблизу кожного обласного центру, багатьох районних центрів і великих міст.

Для захисту людей, які знаходяться в зоні дії деяких радіоелектронних засобів від електромагнітних полів, необхідні організаційні, інженерно-технічні та лікувально-профілактичні заходи.

Санітарні норми та правила для радіотехнічних і електротехнічних об'єктів розроблені на основі медико-біологічних досліджень. Вони контролюють умови їхньої експлуатації, щоб захистити людей від шкідливих електромагнітних випромінювань.

Таким чином, на етапі проєктування об'єкти мають бути розташовані таким чином, щоб інтенсивність опромінення була мінімальною. Крім того, необхідно заздалегідь забезпечити зменшення кількості часу, протягом якого персонал залишається в зоні опромінення. Джерела випромінювання повинні мати мінімальну потужність. Крім того, державні органи повинні дотримуватися державних правил України та не порушувати їх.

ВИСНОВКИ

Нині мобільні технології є невід'ємною частиною людського життя. Вони застосовуються в багатьох галузях: медицина, соціальних, економічних і військових сферах. Застосування мобільних технологій дає змогу істотно знизити часові та матеріальні витрати.

У даній кваліфікаційній роботі була проведена глибока аналітична та дослідницька робота щодо алгоритмічного програмного забезпечення для передачі даних через бездротові децентралізовані мережі. Дослідження включало аналіз та огляд існуючих технологій та алгоритмів, які допомагають досягти оптимальної передачі даних в таких умовах.

В ході нашої роботи було виявлено, що передача даних через бездротові децентралізовані мережі має свої унікальні виклики, такі як динамічна топологія мережі, обмежені ресурси та вплив завищеної перешкоди на сигнал. Для вирішення цих викликів важливо розробляти та вдосконалювати алгоритми маршрутизації, керування доступом та алгоритми оптимізації пропускної здатності.

Кваліфікаційна робота також демонструє значення алгоритмічного програмного забезпечення для підтримки передачі даних через бездротові децентралізовані мережі в різних сферах, включаючи телекомунікації, Інтернет речей (IoT), медичні додатки та багато інших галузей. Підводячи підсумок, можна зазначити, що ця робота підкреслює важливість подальшого дослідження та розвитку алгоритмічного програмного забезпечення для бездротових децентралізованих мереж, з метою досягнення більшої надійності, продуктивності та забезпечення покращеної якості обслуговування в цих мережах.

У результаті виконання кваліфікаційної роботи досягнення поставленої мети було шляхом виконання поставлених завдань:

- Знайдено та проаналізовано наявні аналоги.
- Проаналізовано існуючі протоколи маршрутизації у мережах MANET.

- Спроєктовано алгоритми для передачі даних за допомогою технології Wi-Fi Direct.
- Програмно реалізовано спроєктовані алгоритми.
- Протестовано отримане програмне забезпечення.

У цій роботі досліджувалися протоколи маршрутизації для мобільних бездротових мереж. Попри те, що бездротові мережі активно розвиваються, більшість протоколів перебувають лише в експериментальному вигляді.

Було спроєктовано такі алгоритми: алгоритм пошуку маршруту, алгоритм передавання даних, алгоритм фрагментації даних.

Було спроєктовано структури даних для пакетів у різних ситуаціях, як-от: пошук маршруту, запит маршруту, відповідь на запит маршруту, передавання даних, помилка передавання даних, підтвердження передавання даних.

Розроблено мобільний додаток, який здійснює передачу даних між мобільними пристроями, використовуючи технологію Wi-Fi Direct.

Також були розглянуті аспекти, пов'язані з безпекою праці на робочому місці, включно зі шкідливими та небезпечними факторами, режимами роботи та захистом у надзвичайних ситуаціях.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. X. Guo, S. Yang, L. Cao, J. Wang and Y. Jiang, "A new solution based on optimal link-state routing for named data MANET," in China Communications, vol. 18, no. 4, pp. 213-229.
2. M. Sreenivasu and B. Anil, "Analysis of SEA-based MANET routing protocol for Improved Firefly Energy Optimized Routing," 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2022, pp. 473-477.
3. L. E. Jim and M. A. Gregory, "Improvised MANET Selfish Node Detection using Artificial Immune System based Decision Tree," 2019 29th International Telecommunication Networks and Applications Conference (ITNAC), Auckland, New Zealand, 2019, pp. 1-6.
4. T. Tekdogan, "Analyzing the Traffic of MANETs using Graph Neural Networks," 2022 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT), Soyapango, El Salvador, 2022, pp. 1-4.
5. R. Ramesh and G. Seshikala, "Link Aware Multipath Routing to Defend Against Black Hole Attacks for MANETs," 2023 3rd International Conference on Intelligent Technologies (CONIT), Hubli, India, 2023, pp. 1-6.
6. A. H. Alwan et al., "Monitoring the Impact of the Nodes Density on the MANETs' Environment," 2022 5th International Conference on Engineering Technology and its Applications (IICETA), Al-Najaf, Iraq, 2022, pp. 546-549.
7. W. Wang, B. Yang, X. Wang, Y. She and S. Shen, "Capacity of 3D MANETs Under Packet Redundancy and Receiver Probing," 2018 International Conference on Networking and Network Applications (NaNA), Xi'an, China, 2018, pp. 164-168.

8. A. Gupta and T. Sasikala, "Secure Routing Protocols for MANET-enabled IoT," 2021 IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC), Tumkur, Karnataka, India, 2021, pp. 1-4.
9. I. Nausheen and A. Upadhyay, "An Efficient & Secure Approach under Multiple Attack Prone MANET," 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2023, pp. 686-691.
10. G. Ananthalakshmi and T. Ravindra, "Design and Analysis Of EE-LEACH Routing Protocols in MANET for Wireless Sensor Network," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-4.
11. S. Mohapatra, "Performance analysis of multirate MM-OLSR protocol in wireless adhoc networks," 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 2017, pp. 1494-1499.
12. J. Yin, L. Wang, C. Han and Y. Yang, "NC-OLSR: A network coding based OLSR multipath transmission scheme for FANETs," 2017 4th International Conference on Systems and Informatics (ICSAI), Hangzhou, China, 2017, pp. 1007-1012.
13. Z. Hui and P. S. Yuan, "Analysis and research on OLSR protocol for multi-channel assignment of wireless mesh network," 2017 Chinese Automation Congress (CAC), Jinan, China, 2017, pp. 2732-2737.
14. S. Kakade and P. Khanagoudar, "Performance Analysis of OLSR to Consider Link Quality of OLSR-ETX/MD/ML in Wireless Mesh Networks," 2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS), Bengaluru, India, 2017, pp. 1-10.
15. N. Harrag and A. Harrag, "Bio-inspired OLSR Routing protocol," 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT), Paris, France, 2019, pp. 1763-1767.
16. B. Ardianto, H. S. Dwi Nurcahyo, H. Muftić, R. Harwahyu and R. F. Sari, "Performance Comparison of AODV, AODV-ETX and Modified AODV-ETX in

VANET using NS3," 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), Malang, Indonesia, 2022, pp. 156-161.

17. S. Deepika, N. Nishanth and A. Mujeeb, "An Assessment of Recent Advances in AODV Routing Protocol Path Optimization Algorithms for Mobile Ad hoc Networks," 2021 Fourth International Conference on Microelectronics, Signals & Systems (ICMSS), Kollam, India, 2021, pp. 1-6.
18. T. Ahmad, N. S. Nadiah Binti Ismail, M. A. Rahman, Q. Waseem and A. M. Safri, "Hop Restricted-AODV (HR-AODV) Routing and Its Applicability on Different Wireless Channels for Vehicular Network," 2023 IEEE 8th International Conference On Software Engineering and Computer Systems (ICSECS), Penang, Malaysia, 2023, pp. 1-6.
19. R. I. Al-Essa and G. A. Al-Suhail, "Mobility and Transmission Power of AODV Routing Protocol in MANET," 2022 2nd International Conference on Computing and Machine Intelligence (ICMI), Istanbul, Turkey, 2022, pp. 1-5.
20. S. K. Prasad, S. Gupta, R. B. Singh and T. Sharma, "Performance Testing of AODV using Channel Fading for MANETs," 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), Faridabad, India, 2022, pp. 811-816.
21. V. Trivedi, P. Nayak, D. G. Padhan and N. Mohan, "Performance Enhancement of AODV Routing Protocol for MANET Using Genetic Algorithm," 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2021, pp. 975-981.
22. A. M. Yassin and M. A. Azer, "Performance Comparison of AODV and DSDV In Vehicular Ad Hoc Networks," 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), Cairo, Egypt, 2022, pp. 402-405.
23. M. Hosein and K. Ramdass, "Wi-fi Direct Applications within a Post Covid Classroom – Bridging the Gap between Fully Online and Face to Face Learning,"

2022 International Conference on Information Networking (ICOIN), Jeju-si, Korea, Republic of, 2022, pp. 464-469.

24. D. Ron and J. -R. Lee, "Expectation Maximization Based Power-Saving Method in Wi-Fi Direct," in IEEE Access, vol. 8, pp. 158600-158611.
25. E. Yundra, L. Arianto and U. T. Kartini, "Motion Sensing for Wireless Body Area Networks Based on Android Using Wi-Fi Direct Transmission," 2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE), Surabaya, Indonesia, 2020, pp. 1-5.
26. C. Shao, H. Park, H. Roh, W. Lee and H. Kim, "PolarScout: Wi-Fi Interference-Resilient ZigBee Communication via Shell-Shaping," in IEEE/ACM Transactions on Networking, vol. 28, no. 4, pp. 1587-1600.
27. F. Li et al., "A Local Communication System Over Wi-Fi Direct: Implementation and Performance Evaluation," in IEEE Internet of Things Journal, vol. 7, no. 6, pp. 5140-5158.
28. F. Meneghello, N. D. Fabbro, D. Garlisi, I. Tinnirello and M. Rossi, "A CSI Dataset for Wireless Human Sensing on 80 MHz Wi-Fi Channels," in IEEE Communications Magazine, vol. 61, no. 9, pp. 146-152.
29. C. Joshitha, P. Kanakaraja, S. Rooban, B. S. D. R. Prasad, B. G. Rao and S. V. S. Teja, "Design and Implementation of Wi-Fi Enabled Contactless Electronic Stethoscope," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 932-936.
30. J. Marin, M. Bernhardt and T. Riihonen, "Full-Duplex Constant-Envelope Jamceiver and Self-Interference Suppression by Highpass Filter: Experimental Validation for Wi-Fi Security," in IEEE Journal on Selected Areas in Communications, vol. 41, no. 9, pp. 2937-2950.
31. M. Kamal, A. Abaid, S. Abid and S. Shamail, "FACER-AS: An API Usage-based Code Recommendation Tool for Android Studio," 2021 IEEE International Conference on Software Maintenance and Evolution (ICSME), Luxembourg, 2021, pp. 609-613.

32. V. B, H. S, P. V H and Mohana, "Currency and Fake Currency Detection using Machine Learning and Image Processing – An Application for Blind People using Android Studio," 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2022, pp. 274-277.
33. Y. W. Syaifudin, N. Funabiki, M. S. F. Ruslan, D. C. Wijaya, I. Mu'aasyiqiin and A. R. Patta, "Implementations of Two Answer Submission Methods for Reducing Errors in Android Programming Learning Assistance System," 2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Zallaq, Bahrain, 2021, pp. 126-130.
34. D. Buddhi, R. Singh and A. Gehlot, "Online Virtual Classroom Application In Android Studio," 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC), Bengaluru, India, 2022, pp. 960-963.
35. R. Sharma, T. U. Bux, B. Varshney and K. Tomar, "Real-time Student Management Application Using Google Firebase and Android Studio," 2021 International Conference on Intelligent Technologies (CONIT), Hubli, India, 2021, pp. 1-6.
36. K. Nayak, K. Kushwaha, K. Kumar and J. S. Kumar, "Android based advanced bus reservation system in the pandemic of Covid-19," 2022 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), BHOPAL, India, 2022, pp. 1-6.
37. C. Jiang, Y. Yang and B. Han, "Design and Implementation of Android-based Smart Classroom Control System," 2022 International Conference on Automation, Robotics and Computer Engineering (ICARCE), Wuhan, China, 2022, pp. 1-5.
38. Madona S. Wambua, Modern Android 13 Development Cookbook: Over 70 recipes to solve Android development issues and create better apps with Kotlin and Jetpack Compose , Packt Publishing, 2023.P. Rana, V. Semwal and D. Kalra, "Kreeda: An Android Application for Searching and Organizing Sports Events," 2023 International Conference for Advancement in Technology (ICONAT), Goa, India, 2023, pp. 1-6.

39. ДСанПіН 3.3.2-007-98 «Гігієнічні вимоги до організації роботи з візу-альними дисплейними терміналами електронно-обчислювальних машин».
40. ГН 3.3.5-8-6.6.1-2002 «Гігієнічна класифікація праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу».
41. ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих при-міщень».
42. ДБН В.2.5-28-2006 «Інженерне обладнання будинків і споруд. При-родне і штучне освітлення».
43. НПАОП 0.00-1.29-97 «Правила захисту від статичної електрики».
44. ДСТУ 12.1.005-88 «ССБП. Загальні санітарно-гігієнічні вимоги до повітря робочої зони».
45. ДСТУ Б В.2.5-82:2016 «Електробезпека в будівлях і спорудах. Вимо-ги до захисних заходів від ураження електричним струмом».
46. ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги».
47. НАПБ А.01.001-2004 «Правила пожежної безпеки в Україні».
48. Прогнозування екологічних ризиків з використанням аналізу ієрархій і теорії нечітких множин: Міжнародна науково-практична конференція "І-а Всеукраїнська нарада екологів": Тези доповідей. Україна, м. Вінниця, 4-7 жовтня 2016 року. - 2016. - С.25.
49. Антипов В.В., Давидов Б.І., Тихончук В.С. Біологічна дія, нормування та захист від електромагнітних випромінювань.: Енергоатоміздат, 2012. - 177 с.
50. Філіппов Є.С. Вплив електромагнітних полів на біологічні об'єкти. - 2018. - №1 - Том: 24. - С. 15-19.
51. Екологія та охорона навколишнього природного середовища: навч. посібник для вузів / В. С. Джигирей. - 6-те вид., випр. і доп. - К. : Знання, 2017. - 422 с.
52. Боротьба з шумом на виробництві: Довідник / Під ред. О. Я. Юдіна. – М: Машинобудування, 2015. – 297 с.