

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВОЛИНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ЛЕСІ УКРАЇНКИ
Кафедра комп'ютерних наук та кібербезпеки

На правах рукопису

ТКАЧУК БОГДАН РОМАНОВИЧ
РОЗРОБКА iOS-ДОДАТКУ ДЛЯ МОДЕЛЮВАННЯ ПОШИРЕННЯ
WiFi-СИГНАЛУ

Спеціальність: 122 Комп'ютерні науки
Освітньо-професійна програма: Комп'ютерні науки та інформаційні технології
Кваліфікаційна робота на здобуття освітнього ступеня «магістр»

Науковий керівник:
Булатецький Віталій Вікторович,
Доцент кафедри комп'ютерних наук
та кібербезпеки, кандидат
фізико-математичних наук.

РЕКОМЕНДОВАНО ДО ЗАХИСТУ
Протокол № _____ засідання кафедри
комп'ютерних наук та кібербезпеки
від _____ 2025 р.
Завідувач кафедри
(_____) Гришанович Т. О.

ЛУЦЬК – 2025

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1 МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПОШИРЕННЯ WiFi-СИГНАЛІВ ТА АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ	6
1.1. Теоретичні основи поширення радіосигналів у приміщеннях	6
1.1.2 Модель втрат у вільному просторі (Free Space Path Loss).....	7
1.1.3 Модель Log-Distance Path Loss та її переваги для закритих приміщень	7
1.1.4 Вплив перешкод на поширення сигналу	8
1.1.5 Коефіцієнти атенуації різних будівельних матеріалів	9
1.2 Аналіз існуючих рішень для планування бездротових мереж	10
1.3. Технології розробки мобільних застосунків для платформи iOS.....	16
1.3.1 Характеристики мови програмування Swift та її еволюція.....	16
1.3.2 Декларативний фреймворк SwiftUI для побудови користувацьких інтерфейсів	17
1.3.3 Архітектурний патерн MVVM та його реалізація у SwiftUI-додатках	18
1.3.4 Технологія Core Graphics для візуалізації графічних даних.....	19
1.3.5 Фреймворк Combine для реактивного програмування	21
1.3.6 Система збереження та керування даними у мобільних додатках	21
РОЗДІЛ 2 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ iOS-ДОДАТКУ ДЛЯ МОДЕЛЮВАННЯ БЕЗДРотовИХ МЕРЕЖ	23
2.1 Постановка задачі та функціональні вимоги	23
2.2 Методологія дослідження та життєвий цикл розробки	26
2.3 Теоретичні аспекти математичного моделювання.....	29
2.4 Обґрунтування вибору інструментальних засобів	30
2.5 Етапи програмної реалізації системи	32
2.6 Організація тестування та налагодження програмного засобу	35
2.7 Аналіз отриманих результатів дослідження та рекомендації щодо використання	41
ВИСНОВКИ	43
СПИСОК ДЖЕРЕЛ	46
ДОДАТКИ	51

ВСТУП

Актуальність теми. У сучасному світі бездротові технології стали невід’ємною частиною повсякденного життя та професійної діяльності. За даними Міжнародного союзу електрозв’язку, кількість підключених до Інтернету пристроїв перевищила 5,2 мільярда у 2023 році, при цьому більше 80% з них використовують WiFi підключення [28]. Ефективне планування та оптимізація покриття WiFi сигналу є критично важливими завданнями для забезпечення якісного бездротового зв’язку в житлових, офісних та промислових приміщеннях.

Традиційні методи планування WiFi-мереж часто базуються на емпіричному підході або використанні дорогого професійного обладнання та програмного забезпечення [44]. Більшість існуючих рішень, таких як Ekahau Pro та iBwave Design, орієнтована на корпоративних користувачів і характеризується високою вартістю ліцензування, складним інтерфейсом та потребою у спеціальній підготовці [23]. Водночас, зростає попит на доступні та інтуїтивні інструменти для моделювання поширення WiFi-сигналу, особливо серед IT-спеціалістів малого та середнього бізнесу, студентів телекомунікаційних спеціальностей та домашніх користувачів.

Мобільні пристрої на базі iOS, завдяки своїй обчислювальній потужності та широкій поширеності, представляють ідеальну платформу для розробки доступних інструментів моделювання мереж [42]. За статистикою Apple, станом на 2024 рік використовується понад 1,4 мільярда пристроїв iPhone та iPad по всьому світу [4]. Сучасні можливості мови програмування Swift та декларативного фреймворку SwiftUI дозволяють створювати високопродуктивні додатки з інтуїтивним користувацьким інтерфейсом та складними обчислювальними алгоритмами [6].

Аналіз ринку виявляє значний розрив між професійними та споживчими рішеннями для планування WiFi-покриття. Простіші інструменти не

забезпечують достатньої точності для практичного використання, тоді як професійні системи є неприйнятно дорогими та складними для значної частини потенційних користувачів [15]. Це створює нішу для мобільного рішення, яке поєднуватиме науково обґрунтовані алгоритми розрахунку з доступністю та простотою використання.

Метою роботи є розробка iOS-додатку для моделювання поширення WiFi-сигналу в приміщеннях з використанням мови програмування Swift, який забезпечить доступний та ефективний інструмент для планування бездротових мереж на основі математично обґрунтованих алгоритмів розрахунку.

Для досягнення поставленої мети необхідно:

- проаналізувати теоретичні основи поширення радіосигналів у закритих приміщеннях та дослідити математичні моделі розрахунку втрат сигналу з урахуванням перешкод різних типів;
- дослідити існуючі рішення для планування WiFi мереж, виконати порівняльний аналіз їх функціональних можливостей та визначити вимоги до розробки мобільного інструменту;
- обґрунтувати вибір технологій та архітектурних рішень для розробки iOS-додатку з урахуванням специфіки мобільної платформи та вимог до продуктивності;
- розробити математичну модель поширення WiFi-сигналу на основі алгоритму Log-Distance Path Loss з урахуванням атенуації від будівельних матеріалів різних типів;
- спроектувати та реалізувати архітектуру iOS-додатку з використанням патерну MVVM;
- створити алгоритми візуалізації результатів моделювання у вигляді інтерактивних теплових карт та функціональність експорту результатів у різних форматах;
- провести валідацію точності математичної моделі та тестування функціональності розробленого додатку на практичних сценаріях використання.

Об'єктом дослідження є процеси поширення електромагнітних хвиль WiFi-діапазону в закритих приміщеннях з урахуванням впливу архітектурних елементів та будівельних матеріалів на характеристики покриття бездротової мережі.

Предметом дослідження є математичні моделі, алгоритми розрахунку та методи програмної реалізації систем моделювання поширення WiFi-сигналу для мобільних пристроїв на платформі iOS з використанням мови програмування Swift.

Наукова новизна роботи полягає у розробці підходу до реалізації математичного моделювання поширення бездротових сигналів на мобільних пристроях з використанням сучасних технологій iOS-розробки. Вперше створено повнофункціональний мобільний інструмент для моделювання WiFi-покриття, який поєднує науково обґрунтовану математичну модель з інтуїтивним сенсорним інтерфейсом та можливостями інтерактивної візуалізації результатів у режимі реального часу.

Практичне застосування розробленого рішення полягає у забезпеченні доступного інструменту для планування WiFi-мереж широкому колу користувачів, включаючи IT-спеціалістів малого та середнього бізнесу, студентів телекомунікаційних спеціальностей, викладачів навчальних закладів та домашніх користувачів. Додаток забезпечує зниження часових та фінансових витрат на планування бездротових мереж, підвищує якість покриття через використання науково обґрунтованих методів розрахунку та сприяє популяризації знань про принципи радіотехніки серед широкого кола користувачів.

РОЗДІЛ 1

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПОШИРЕННЯ WiFi-СИГНАЛІВ ТА АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ

1.1. Теоретичні основи поширення радіосигналів у приміщеннях

1.1.1 Фізичні принципи розповсюдження електромагнітних хвиль

Поширення WiFi-сигналів у закритих приміщеннях підпорядковується фундаментальним законам електродинаміки та характеризується складною взаємодією електромагнітних хвиль з архітектурними елементами будівель [16]. На відміну від ідеальних умов вільного простору, розповсюдження радіохвиль у приміщеннях супроводжується численними фізичними процесами, що значно впливають на характеристики сигналу.

Основними механізмами втрат енергії електромагнітного поля є відбиття від поверхонь з різними діелектричними властивостями, поглинання енергії матеріалами будівельних конструкцій, розсіювання на неоднорідностях середовища та дифракція навколо перешкод [24]. Частота сигналу визначає глибину проникнення в матеріали та ступінь взаємодії з об'єктами певних розмірів. Для стандартних WiFi-частот 2.4 ГГц та 5 ГГц характерні різні закономірності поширення у будівельних матеріалах.

Багатопроменеве поширення створює складну інтерференційну картину, коли прямий сигнал від передавача комбінується з відбитими компонентами від стін, стелі та інших поверхонь [45]. Цей ефект може як посилювати сигнал у певних точках простору завдяки конструктивній інтерференції, так і створювати зони глибокого завмирання через деструктивне накладання хвиль.

Поляризаційні ефекти впливають на ефективність передачі енергії між антенами передавача та приймача. Зміна орієнтації антен або наявність металевих об'єктів може призводити до деполяризації сигналу та додаткових втрат потужності [3].

1.1.2 Модель втрат у вільному просторі (Free Space Path Loss)

Базовою моделлю для розрахунку втрат потужності радіосигналу є формула втрат у вільному просторі, що описує ідеальні умови розповсюдження без перешкод [21]. Ця модель застосовується як референтна точка для більш складних моделей, що враховують реальні умови приміщень.

Загальна формула втрат у вільному просторі має вигляд:

$$FSPL(dB) = 20 \log_{10}(d) + 20 \log_{10}(f) + 20 \log_{10}(4\pi/c)$$

Де d позначає відстань у метрах між передавачем та приймачем, f представляє частоту сигналу у герцах, а c означає швидкість світла, що дорівнює 3×10^8 метрів за секунду.

Для практичних розрахунків у WiFi-діапазоні формула спрощується до зручнішого вигляду:

$$FSPL(dB) = 20 \log_{10}(d) + 20 \log_{10}(f_MHz) - 27.55$$

Застосовуючи цю формулу до стандартних WiFi-частот, отримуємо специфічні вирази для кожного діапазону. Для частоти 2.4 ГГц втрати обчислюються як $FSPL = 20 \log_{10}(d) + 40.05$.

Для 5 ГГц діапазону формула набуває вигляду:

$$FSPL = 20 \log_{10}(d) + 46.5 [29].$$

Різниця у втратах між діапазонами становить приблизно 6.4 дБ на користь нижчої частоти, що пояснює кращу проникність сигналів 2.4 ГГц через стіни та інші перешкоди. Проте ця модель не враховує реальних умов приміщень та потребує подальшого удосконалення.

1.1.3 Модель Log-Distance Path Loss та її переваги для закритих приміщень

Модель Log-Distance Path Loss представляє собою розширення базової моделі втрат у вільному просторі, адаптоване для реальних умов розповсюдження радіохвиль у закритих приміщеннях [30]. Ця модель введена Міжнародним союзом електрозв'язку у рекомендації ITU-R P.1238 як стандартний метод прогнозування втрат сигналу у внутрішніх радіосистемах.

Математичне вираження моделі має вигляд:

$$PL(d) = PL(d_0) + 10n \log_{10}(d/d_0) + X_{\sigma}$$

Де $PL(d)$ означає втрати сигналу на відстані d , $PL(d_0)$ представляє втрати на опорній відстані d_0 , n є показником втрат шляху, що характеризує швидкість загасання сигналу, а X_{σ} описує випадкову компоненту затінення.

Опорна відстань d_0 зазвичай встановлюється на рівні одного метра, що відповідає стандартним методикам вимірювання характеристик антен. Для цієї відстані втрати розраховуються за формулою вільного простору [7].

Показник втрат шляху n є ключовим параметром моделі, що визначає швидкість загасання сигналу зі збільшенням відстані. Для відкритого простору цей показник дорівнює 2.0, що відповідає квадратичній залежності втрат від відстані. У реальних приміщеннях значення n зазвичай перевищує 2,0 через наявність перешкод та багатопроменеве розповсюдження.

Типові значення показника втрат шляху для різних типів приміщень складають: для відкритого простору $n = 2.0$, для офісних приміщень з невеликою кількістю перешкод $n = 2.2-2.8$, для житлових приміщень $n = 2.8-3.2$, а для промислових об'єктів з великою кількістю металевих конструкцій $n = 3.0-4.0$ [32].

Компонента затінення X_{σ} моделює випадкові варіації сили сигналу, спричинені локальними неоднорідностями середовища. Вона описується нормальним розподілом з нульовим математичним сподіванням та стандартним відхиленням σ , що зазвичай становить 4-12 дБ залежно від типу приміщення.

1.1.4 Вплив перешкод на поширення сигналу

Архітектурні елементи будівель створюють значні перешкоди для розповсюдження WiFi-сигналів, спричиняючи додаткові втрати потужності через поглинання та відбиття електромагнітної енергії [25]. Характер взаємодії сигналу з перешкодами залежить від типу матеріалу, товщини конструкції, частоти сигналу та кута падіння електромагнітної хвилі.

Механізм втрат від перешкод моделюється через додаткові компоненти в загальній формулі розрахунку втрат сигналу. Загальна формула втрат з урахуванням стін має вигляд:

$$PL_total(d) = PL(d) + \Sigma WAF_i,$$

де ΣWAF_i означає суму коефіцієнтів атенуації для всіх стін на шляху сигналу, а WAF_i представляє коефіцієнт атенуації для кожної окремої стіни [14].

Розрахунок впливу перешкод потребує визначення геометричного шляху сигналу від передавача до точки прийому та ідентифікації всіх стін, що перетинають цей шлях. Алгоритм обчислення включає побудову прямої лінії між джерелом сигналу та точкою спостереження, виявлення перетинів цієї лінії з усіма стінами на плані приміщення, визначення типу матеріалу кожної стіни та сумування відповідних коефіцієнтів атенуації.

1.1.5 Коефіцієнти атенуації різних будівельних матеріалів

Будівельні матеріали характеризуються різними діелектричними властивостями, що визначають ступінь поглинання та відбиття електромагнітної енергії WiFi-діапазону [48]. Коефіцієнти атенуації матеріалів визначаються експериментально та стандартизовані у міжнародних рекомендаціях.

Гіпсокартонні перегородки демонструють мінімальну атенуацію на рівні 3-5 дБ завдяки низькому вмісту води та відносно невеликій товщині. Ці конструкції широко застосовуються у сучасному офісному будівництві та забезпечують найкращу прозорість для WiFi-сигналів [35].

Дерев'яні конструкції створюють помірні втрати на рівні 4-6 дБ, що залежить від породи деревини, вологості матеріалу та товщини елемента. Вологе дерево демонструє значно вищу атенуацію через збільшення діелектричних втрат.

Цегляні стіни характеризуються атенуацією 6-8 дБ та широко представлені у житловій забудові. Точне значення втрат залежить від типу цегли, товщини кладки та наявності металевої арматури.

Бетонні конструкції створюють найбільші втрати серед типових будівельних матеріалів на рівні 10-15 дБ через високу щільність матеріалу та часто наявність металевого армування. Залізобетонні стіни можуть практично повністю блокувати WiFi-сигнал [49].

Скляні перегородки демонструють відносно низьку атенуацію 2-3 дБ для звичайного скла, проте металізовані або енергозберігаючі покриття можуть значно збільшувати втрати до 20-30 дБ.

Металеві конструкції створюють найвищі втрати понад 20 дБ та можуть повністю блокувати проходження сигналу. Такі матеріали фактично працюють як екрани Фарадея для радіочастотних сигналів.

1.2 Аналіз існуючих рішень для планування бездротових мереж

Ринок професійного програмного забезпечення для планування WiFi-мереж характеризується широким спектром рішень, що охоплюють різні сегменти користувачів від індивідуальних консультантів до великих корпорацій [17]. Сучасні інструменти планування можна класифікувати за функціональною складністю, цільовою аудиторією та вартістю ліцензування.

EkaHau Pro займає лідируючі позиції у сегменті професійного планування WiFi-мереж та широко визнаний як галузевий стандарт [43]. Продукт забезпечує повний цикл проектування бездротових мереж від початкового планування до пост-інсталяційної валідації та оптимізації. Система використовує власну математичну модель поширення сигналу, що враховує багатопроменеве розповсюдження, інтерференцію та детальні характеристики радіообладнання. Основні переваги включають високу точність прогнозування покриття з похибкою менше 5 дБ у 90% випадків, повну підтримку сучасних стандартів WiFi включаючи 802.11ax, розширені можливості автоматизованого site survey та інтеграцію з професійними CAD-системами. Вартість ліцензії EkaHau Pro становить від 3000 до 5000 доларів річно, що робить його доступним переважно для великих системних інтеграторів та корпоративних користувачів.

iBwave Wi-Fi орієнтований на корпоративний сегмент та великі інфраструктурні об'єкти з особливими вимогами до точності моделювання [18]. Система забезпечує тривимірне моделювання поширення сигналу з урахуванням вертикальної складової, що критично важливо для багатоповерхових будівель. Унікальною особливістю є підтримка автоматичного розміщення точок доступу з використанням алгоритмів оптимізації, розрахунок міжканальної інтерференції та динамічну оптимізацію частотного планування. Продукт інтегрується з системами керування будівлями та забезпечує експорт конфігурацій безпосередньо до контролерів WiFi провідних виробників.

NetSpot представляє середній сегмент ринку, поєднуючи доступну ціну з професійними можливостями планування [41]. Продукт доступний для операційних систем Windows та macOS і забезпечує базовий функціонал планування мереж та site survey. Вартість ліцензії варіюється від 149 до 499 доларів залежно від комплектації, що робить його привабливим для малого та середнього бізнесу.

WiFi Explorer представляє спеціалізований інструмент для аналізу радіочастотного середовища та оптимізації існуючих мереж [50]. Продукт фокусується на детальному аналізі спектра, виявленні джерел інтерференції та рекомендаціях щодо частотного планування.

Детальний аналіз функціональних характеристик провідних рішень виявляє значні відмінності у можливостях та підходах до моделювання покриття мереж [11]. Точність математичного моделювання варіюється від простих моделей втрат у вільному просторі до складних алгоритмів, що враховують багатопроточне поширення, дифракційні ефекти та статистичні варіації сигналу.

Професійні інструменти на кшталт Ekahau Pro та iBwave використовують складні математичні моделі, що поєднують детерміністичні розрахунки з емпіричними поправками [31]. Ці системи враховують частотно-залежні характеристики поширення, поляризаційні ефекти, багатопроточне відбиття та

статистичні флуктуації сигналу. Точність прогнозування досягає 5-8 дБ середньої абсолютної похибки у типових офісних середовищах.

Інтеграція з CAD-системами є критично важливою функціональністю для професійних інструментів [10]. Високоякісні системи підтримують імпорту архітектурних планів у стандартних форматах DWG, DXF та PDF, що значно спрощує процес створення точних тривимірних моделей приміщень. Автоматичне розпізнавання будівельних елементів та присвоєння відповідних радіочастотних характеристик прискорює процес підготовки моделі.

Тривимірне моделювання стає обов'язковою вимогою для планування мереж у багатоповерхових будівлях [18]. Сучасні рішення враховують вертикальну складову поширення сигналу, що дозволяє оптимізувати розміщення антен не тільки у горизонтальній площині, але й за висотою. Це особливо важливо для великих торгових центрів, офісних комплексів та промислових об'єктів.

Автоматизована оптимізація розміщення точок доступу використовує алгоритми генетичної оптимізації, метод відпалу та інші евристичні підходи [34]. Системи автоматично визначають оптимальні позиції для антен з урахуванням вимог до покриття, обмежень на потужність сигналу та мінімізації інтерференції між каналами.

Функціональність site survey інтегрує планування з валідацією через підтримку спеціалізованого вимірювального обладнання [31]. Професійні інструменти забезпечують порівняння прогнозованих та вимірених значень, автоматичне калібрування математичних моделей та генерацію звітів про відповідність проекту реальним характеристикам мережі.

Комплексний аналіз наявних рішень виявляє характерний розподіл переваг та обмежень залежно від цінового сегменту та цільової аудиторії [15]. Професійні системи демонструють суттєві переваги у точності моделювання та повноті функціонального забезпечення, проте характеризуються значними бар'єрами для входу у вигляді високої вартості та складності освоєння.

Переваги професійних систем включають високу точність математичних розрахунків з середньою похибкою 5-8 дБ, що забезпечується через використання складних алгоритмів багатопроменевого моделювання та емпіричну калібровку [31]. Повна підтримка сучасних стандартів WiFi охоплює всі діапазони від 2.4 ГГц до 6 ГГц включаючи найновіші технології 802.11ax та 802.11be. Розширені можливості зв'язності забезпечують генерацію професійної документації з детальними характеристиками покриття, аналізом інтерференції та рекомендаціями щодо оптимізації.

Інтеграція з корпоративними системами дозволяє експорт конфігурацій безпосередньо до контролерів WiFi провідних виробників, автоматичну синхронізацію з системами керування мережею та інтеграцію з базами даних обладнання [18]. Підтримка складних сценаріїв використання включає моделювання мереж з тисячами точок доступу, оптимізацію покриття для специфічних додатків та аналіз продуктивності в умовах високої щільності користувачів.

Недоліки професійних систем створюють значні бар'єри для широкого впровадження [15]. Висока вартість ліцензування від 3000 до 10000 доларів річно робить ці рішення недоступними для малого та середнього бізнесу. Складність освоєння та використання потребує спеціалізованого навчання та досвіду роботи з радіочастотними технологіями. Надлишковий функціонал для простих задач створює непотрібну складність при плануванні невеликих мереж домашнього або офісного типу.

Прив'язка до настільних платформ обмежує мобільність інженерів та унеможлиблює проведення оперативного планування безпосередньо на об'єкті [43]. Потреба у потужному апаратному забезпеченні та тривалий час розрахунків знижують продуктивність роботи з великими проектами.

Характеристики простіших рішень демонструють протилежний набір переваг та обмежень [41]. Доступна вартість або безкоштовність робить ці інструменти привабливими для широкого кола користувачів. Простий інтерфейс та швидке освоєння не потребують спеціального навчання або глибоких знань

радіотехніки. Обмежена функціональність фокусується на основних завданнях планування без надлишкової складності.

Проте ці рішення характеризуються недостатньою точністю математичного моделювання через використання спрощених алгоритмів [50]. Обмежена підтримка сучасних стандартів WiFi та відсутність функцій оптимізації знижують практичну цінність результатів планування. Мінімальні можливості звітності не забезпечують генерацію професійної документації, необхідної для великих проектів.

Сегмент мобільних додатків для планування WiFi-мереж залишається недостатньо розвиненим порівняно з настільними рішеннями [50]. Більшість наявних мобільних інструментів обмежуються функціями аналізу існуючих мереж та простими калькуляторами покриття без можливості повноцінного планування нових інсталяцій.

WiFi Analyzer для Android забезпечує базовий аналіз радіочастотного спектра та виявлення доступних мереж, проте не включає функції моделювання покриття [50]. Network Analyzer для iOS надає інформацію про параметри підключення та швидкість передачі даних, але не підтримує планування майбутніх інсталяцій.

Основні обмеження існуючих мобільних рішень включають відсутність складних математичних моделей через обмеження обчислювальної потужності мобільних процесорів попередніх поколінь [42]. Спрощені інтерфейси не забезпечують функціональність, необхідну для професійного планування мереж. Відсутність інтеграції з CAD-системами ускладнює створення точних моделей приміщень.

Обмежені можливості експорту не дозволяють генерувати професійну документацію та інтегрувати результати з існуючими робочими процесами. Фокус на аналізі існуючих мереж замість планування нових інсталяцій знижує практичну цінність для проектувальників.

Аналіз ринку виявляє значний розрив між професійними та споживчими рішеннями для планування WiFi-покриття [50]. Більшість простих інструментів

не забезпечують достатньої точності для практичного використання через спрощені математичні моделі. Професійні системи є неприйнятно дорогими та складними для значної частини потенційних користувачів, включаючи ІТ-спеціалістів малого та середнього бізнесу, студентів телекомунікаційних спеціальностей та домашніх користувачів.

Мобільні платформи представляють унікальну можливість для створення доступного та зручного інструменту планування [42]. Сучасні смартфони та планшети володіють достатньою обчислювальною потужністю для виконання складних математичних розрахунків у режимі реального часу. Багатоядерні процесори з тактовими частотами понад 3 ГГц та обсягом оперативної пам'яті 8-16 ГБ забезпечують продуктивність, порівнянну з настільними комп'ютерами минулих поколінь.

Інтуїтивні сенсорні інтерфейси мобільних пристроїв дозволяють створювати зручні інструменти для інтерактивного планування [38]. Жести багатодотикового управління забезпечують природну взаємодію з планами приміщень, а високороздільні дисплеї дозволяють деталізовано візуалізувати результати моделювання.

Ключові переваги мобільного підходу включають доступність та портативність, що дозволяє проводити планування безпосередньо на об'єкті інсталяції. Інтуїтивний touch-інтерфейс спрощує процес малювання планів приміщень та розміщення обладнання. Можливість проведення вимірювань та верифікації прогнозів безпосередньо на місці підвищує точність планування.

Низька вартість входу для користувачів через безкоштовне або недороге розповсюдження через магазини додатків робить професійні інструменти планування доступними широкій аудиторії [4]. Простота поширення та оновлення через централізовані магазини забезпечує швидке впровадження нових функцій та виправлень помилок.

Інтеграція з екосистемою мобільного пристрою дозволяє використовувати вбудовані сенсори для покращення точності моделювання [42]. GPS для геолокації, компас для орієнтації планів, камера для створення фотопланів

приміщень та інші датчики можуть суттєво розширити функціональність планувальника.

1.3. Технології розробки мобільних застосунків для платформи iOS

1.3.1 Характеристики мови програмування Swift та її еволюція

Мова програмування Swift, представлена корпорацією Apple у 2014 році, стала результатом еволюції підходів до розробки програмного забезпечення для мобільних пристроїв та є заміною застарілої Objective-C [33]. Swift розроблявся з метою поєднання продуктивності системних мов програмування з виразністю та безпекою високорівневих мов, що робить його ідеальним вибором для створення складних обчислювальних додатків.

Система типів Swift базується на принципах сильної статичної типізації з можливостями виводу типів [6]. Компілятор автоматично визначає типи змінних на основі присвоюваних значень, що знижує багатослівність коду без втрати безпеки типів. Опціональні типи забезпечують явну обробку можливості відсутності значення, ефективно запобігаючи помилкам доступу до null-посилань, які є поширеною причиною збоїв у додатках.

Автоматичне управління пам'яттю реалізоване через механізм підрахунку посилань ARC [8]. Система автоматично відстежує кількість посилань на об'єкти та звільняє пам'ять, коли об'єкт більше не використовується. Це забезпечує детерміністичне управління пам'яттю без накладних витрат збирача сміття, що критично важливо для мобільних пристроїв з обмеженими ресурсами.

Продуктивність Swift досягається завдяки компіляції у нативний машинний код з використанням оптимізуючого компілятора LLVM [33]. Бенчмарки демонструють, що Swift часто перевершує Objective-C за швидкістю виконання завдяки кращим оптимізаціям компілятора та відсутності overhead від динамічної диспетчеризації методів у критичних ділянках коду.

Сучасні мовні конструкції включають розширені можливості функціонального програмування через функції вищого порядку map, filter,

reduce, підтримку замикань з автоматичним захопленням змінних, pattern matching для елегантної обробки складних структур даних та потужну систему протоколів, що забезпечує гнучку архітектуру без множинного наслідування [33].

Інтероперабельність з Objective-C дозволяє поступово мігрувати існуючі проекти та використовувати багату екосистему існуючих бібліотек iOS [42]. Двостороння сумісність забезпечує можливість викликати Swift-код з Objective-C та навпаки без значних накладних витрат на конверсію типів.

Для завдань математичного моделювання Swift надає нативні типи для роботи з числами подвійної точності Float64 та Double, вбудовані математичні функції та константи через модуль Foundation, ефективні алгоритми сортування та пошуку з оптимізацією для різних типів даних, а також інтеграцію з Grand Central Dispatch для паралелізації обчислень [33].

1.3.2 Декларативний фреймворк SwiftUI для побудови користувацьких інтерфейсів

Фреймворк SwiftUI, представлений у 2019 році, революціонізував підходи до створення користувацьких інтерфейсів на платформах Apple завдяки впровадженню декларативної парадигми програмування [5]. На відміну від традиційного імперативного підходу UIKit, де розробник детально описує кожен крок створення та модифікації інтерфейсу, SwiftUI дозволяє описувати бажаний кінцевий стан інтерфейсу, а система автоматично забезпечує його досягнення та підтримку.

Декларативність означає, що код описує структуру та поведінку інтерфейсу через композицію View-компонентів без необхідності ручного управління життєвим циклом елементів [20]. Наприклад, створення інтерактивного елемента управління силою сигналу може бути реалізоване через декларативний опис структури без явного створення та налаштування об'єктів.

Реактивність забезпечується через систему прив'язки даних з використанням property wrapper'ів @State, @ObservedObject, @Published, які автоматично відстежують зміни у даних та оновлюють відповідні частини

інтерфейсу [9]. Це усуває необхідність ручного управління оновленнями UI та значно знижує ймовірність помилок синхронізації між станом даних та їх представленням.

Композиційність дозволяє створювати складні інтерфейси через комбінування простих, багаторазово використовуваних компонентів [38].

Кожен View-компонент є самодостатнім та може бути легко інтегрований у різні контексти без модифікації внутрішньої логіки. Це сприяє створенню модульних архітектур та спрощує тестування окремих компонентів.

Кроссплатформенність забезпечує можливість використання одного коду для iOS, macOS, watchOS та tvOS з автоматичною адаптацією до специфіки кожної платформи [38]. Система автоматично застосовує відповідні стилі, розміри та поведінку елементів управління залежно від цільової платформи.

Для проекту моделювання WiFi SwiftUI надає Canvas-компоненти для створення інтерактивних областей малювання планів приміщень, систему жестів для обробки торкань, перетягування та масштабування, вбудовану систему анімацій для плавного оновлення візуалізацій при зміні параметрів моделювання, а також адаптивні системи компонування для підтримки різних розмірів екранів від iPhone SE до iPad Pro [20].

1.3.3 Архітектурний патерн MVVM та його реалізація у SwiftUI-додатках

Model-View-ViewModel є рекомендованим архітектурним патерном для SwiftUI-додатків, що забезпечує чітке розділення відповідальності між компонентами системи та оптимальну інтеграцію з реактивною природою фреймворку [19]. Патерн особливо ефективний для складних додатків з багатою бізнес-логікою та динамічними користувацькими інтерфейсами.

Рівень Model інкапсулює всю бізнес-логіку додатку та структури даних [26]. У контексті WiFi-симулятора це включає структури для представлення геометричних об'єктів, математичні алгоритми розрахунку поширення сигналу, логіку валідації вхідних параметрів та механізми серіалізації для збереження та

завантаження проектів. Model є незалежним від представлення та може бути легко протестований у ізоляції.

Рівень View відповідає виключно за представлення даних користувачу та обробку користувацького вводу [22]. У SwiftUI View-компоненти є декларативними структурами, що описують бажаний вигляд інтерфейсу. Для симулятора WiFi це включає інтерфейси для інтерактивного малювання планів приміщень, елементи управління налаштуваннями параметрів моделювання, компоненти візуалізації теплових карт та діалогові вікна експорту результатів.

ViewModel виступає проміжним шаром, що забезпечує взаємодію між View та Model [19]. Основні функції ViewModel включають перетворення даних з Model у формат, зручний для відображення у View, обробку користувацьких команд та їх трансляцію у виклики методів Model, управління асинхронними операціями та їх результатами, а також забезпечення реактивності через @Published властивості, які автоматично повідомляють View про зміни у стані.

Переваги MVVM для складних додатків включають незалежне тестування бізнес-логіки через ізоляцію логіки у ViewModel від деталей представлення [40]. Можливість повторного використання ViewModel для різних View-компонентів забезпечує консистентність поведінки у різних частинах додатку. Чітке розділення відповідальності спрощує розуміння архітектури та полегшує співпрацю у команді розробників.

Реалізація MVVM у SwiftUI використовує ObservableObject протокол для ViewModel класів та @Published property wrapper для реактивних властивостей [9]. View-компоненти підписуються на зміни у ViewModel через @StateObject або @ObservedObject, що забезпечує автоматичне оновлення інтерфейсу при зміні стану без необхідності ручного управління lifecycle компонентів.

1.3.4 Технологія Core Graphics для візуалізації графічних даних

Core Graphics є низькорівневим API для двовимірної векторної графіки, що забезпечує точний контроль над процесом рендерингу та є критично важливим

для створення складних візуалізацій у додатках наукового та технічного спрямування [21]. Фреймворк базується на моделі Quartz 2D та забезпечує кроссплатформенну сумісність між iOS та macOS.

Контексти рендерингу є центральним концептом Core Graphics та представляють абстракцію над різними поверхнями малювання [20]. Додаток може створювати контексти для малювання на екрані, у растрових зображеннях, у PDF-документах або у векторних форматах. Це забезпечує уніфікований підхід до створення візуалізацій та їх експорту у різних форматах без дублювання коду рендерингу.

Векторна графіка забезпечує масштабованість зображень без втрати якості завдяки математичному опису геометричних примітивів замість растрового представлення [21]. Це особливо важливо для підтримки різних роздільностей екранів сучасних мобільних пристроїв та генерації високоякісної документації.

Система кольорових просторів дозволяє точне управління кольоропередачею через підтримку різних кольорових моделей RGB, CMYK, LAB [20]. Градієнти забезпечують створення плавних кольорових переходів, що критично важливо для візуалізації теплових карт з безперервними змінами інтенсивності сигналу.

Афінні трансформації забезпечують математично точне масштабування, поворот, зміщення та деформацію графічних об'єктів [21]. Система трансформацій дозволяє створювати складні інтерактивні інтерфейси з можливістю масштабування та панорамування планів приміщень.

Для проекту моделювання WiFi Core Graphics надає можливості створення інтерактивних канвасів для малювання планів з підтримкою векторних примітивів, генерації теплових карт з використанням радіальних градієнтів та blend modes, експорту результатів у векторні формати PDF для професійної документації, а також оптимізації продуктивності рендерингу через кешування растрових представлень складних графічних елементів [20].

1.3.5 Фреймворк Combine для реактивного програмування

Combine є нативним фреймворком Apple для реактивного програмування, що забезпечує декларативний підхід до обробки асинхронних подій та потоків даних [9]. Фреймворк інтегрований з SwiftUI та забезпечує ефективне управління складними асинхронними операціями у мобільних додатках.

Основні концепції Combine включають Publisher'и, які генерують потоки значень або помилок у часі, Subscriber'и, які отримують та обробляють ці значення, та Operator'и, які трансформують, фільтрують та комбінують потоки даних [9]. Ця архітектура забезпечує композиційний підхід до побудови складних асинхронних систем.

Backpressure management дозволяє контролювати швидкість генерації та споживання даних для запобігання перевантаження пам'яті [9]. Система автоматично регулює швидкість обробки залежно від можливостей споживача.

Інтеграція з SwiftUI реалізована через @Published property wrapper, який автоматично створює Publisher для властивостей ViewModel [9]. Зміни у цих властивостях автоматично транслюються у оновлення інтерфейсу через reactive binding.

Для симулятора WiFi Combine забезпечує реактивне управління змінами параметрів моделювання з автоматичним перерахунком теплових карт, асинхронну обробку складних математичних розрахунків без блокування головного потоку UI, обробку користувацьких жестів та їх трансляцію у зміни моделі з відповідними анімаціями, а також координацію між різними частинами додатку через централізовані потоки подій [9].

1.3.6 Система збереження та керування даними у мобільних додатках

Управління даними у iOS-додатках забезпечується через комбінацію локального збереження та інтеграції з хмарними сервісами [42]. Для додатку моделювання WiFi критично важливими є надійність збереження проєктів,

швидкість доступу до даних та можливість експорту результатів у стандартних форматах.

UserDefaults забезпечує збереження простих налаштувань та переваг користувача через key-value сховище [42]. Система автоматично синхронізується між запусками додатку та забезпечує Thread-safe доступ до даних.

Codable протокол забезпечує серіалізацію складних об'єктних структур у JSON або Property List формати [6]. Автоматична генерація коду серіалізації знижує ймовірність помилок та спрощує підтримку структур даних.

FileManager надає API для роботи з файловою системою iOS з урахуванням обмежень безпеки sandbox [42]. Система забезпечує організацію файлів у відповідних директоріях та автоматичне резервне копіювання через iCloud.

Для WiFi-симулятора система збереження включає JSON-серіалізацію проектів карт з усіма геометричними об'єктами та параметрами, автоматичне збереження стану сесії для відновлення роботи після перезапуску, експорт результатів у стандартних форматах через Share Sheet API, а також інтеграцію з Files app для централізованого управління проектами [42].

РОЗДІЛ 2

ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ iOS-ДОДАТКУ ДЛЯ МОДЕЛЮВАННЯ БЕЗДРОТОВИХ МЕРЕЖ

2.1 Постановка задачі та функціональні вимоги

Формулювання основного завдання дослідження. Основною метою розроблюваної системи є створення мобільного iOS-додатку, що забезпечує точне моделювання поширення WiFi-сигналу в закритих приміщеннях з урахуванням архітектурних особливостей та різних типів будівельних матеріалів [27]. Система має надавати користувачам інтуїтивний інструмент для планування розміщення точок доступу WiFi з подальшою візуалізацією зон покриття у вигляді теплових карт.

Ключові завдання системи включають забезпечення математично обґрунтованого моделювання поширення радіосигналу на основі моделі Log-Distance Path Loss з урахуванням атенуації від перешкод різних типів. Система повинна реалізувати інтерактивний інтерфейс для створення планів приміщень та розміщення джерел сигналу з використанням сенсорних можливостей мобільних пристроїв.

Створення системи візуалізації результатів моделювання у зручному для аналізу форматі передбачає генерацію теплових карт з різними режимами відображення та кольоровими схемами. Забезпечення можливості збереження проектів та експорту результатів у професійних форматах дозволить інтегрувати систему в існуючі робочі процеси планування мереж.

Функціональні вимоги до системи моделювання. Система управління картами та проектами повинна забезпечувати створення нових карт приміщень з можливістю задання назви та масштабу для точного відображення реальних розмірів об'єктів. Збереження та завантаження проектів у локальному сховищі пристрою гарантує доступність даних без потреби в постійному інтернет-з'єднанні. Функціональність видалення непотрібних карт з можливістю підтвердження дії запобігає випадковій втраті важливих даних.

Підсистема малювання та редагування планів має забезпечувати інтерактивне малювання стін за допомогою сенсорного інтерфейсу з використанням природних жестів торкання та перетягування. Вибір типу матеріалу стіни з предефінованого списку включає гіпсокартон, цеглу, бетон, дерево та скло з відповідними коефіцієнтами атенуації. Можливість видалення окремих стін або повна очистка карти забезпечує гнучкість редагування проектів.

Функціональність роботи з джерелами WiFi передбачає розміщення точок доступу на карті з можливістю налаштування параметрів потужності передачі, частотного діапазону та типу антени. Система повинна підтримувати множинні джерела сигналу з індивідуальними налаштуваннями для кожного. Можливість тимчасового відключення джерел без видалення дозволяє експериментувати з різними конфігураціями.

Підсистема моделювання та візуалізації має реалізовувати розрахунок поширення сигналу в режимі реального часу з відображенням результатів у вигляді теплових карт. Система повинна підтримувати різні режими візуалізації включаючи силу сигналу, зони покриття, втрати шляху та інтерференцію між джерелами.

Нефункціональні вимоги до продуктивності та зручності використання. Вимоги до продуктивності визначають, що час розрахунку теплової карти для приміщення середнього розміру не повинен перевищувати 500 мілісекунд для забезпечення плавного користувацького досвіду. Споживання оперативної пам'яті має бути оптимізовано для роботи на пристроях з обмеженими ресурсами починаючи від iPhone 8 з 2 ГБ RAM.

Вимоги до сумісності передбачають підтримку iOS версії 16.0 та новіших для забезпечення доступу до сучасних API та функцій безпеки. Додаток повинен коректно працювати на всіх сучасних моделях iPhone та iPad з автоматичною адаптацією інтерфейсу до різних розмірів екранів.

Критерії зручності використання включають інтуїтивність інтерфейсу, що дозволяє новим користувачам освоїти основні функції протягом 15 хвилин без

додаткового навчання. Час відгуку на користувацькі дії не повинен перевищувати 100 мілісекунд для забезпечення відчуття миттєвої реакції системи.

Вимоги до надійності визначають, що додаток повинен коректно обробляти помилки вводу та нештатні ситуації без аварійного завершення роботи. Автоматичне збереження стану роботи має відбуватися кожні 2 хвилини для запобігання втраті даних при несподіваному закритті додатку.

Обмеження та припущення при розробці. Основні обмеження розробки пов'язані з апаратними можливостями мобільних пристроїв та особливостями платформи iOS. Обчислювальна потужність мобільних процесорів накладає обмеження на складність математичних моделей та розмір сітки розрахунку для забезпечення прийнятної швидкодії.

Обмеження пам'яті вимагають ефективного управління ресурсами та оптимізації алгоритмів для роботи з великими масивами даних теплових карт. Розмір додатку повинен залишатися у межах 50-100 МБ для зручності завантаження через мобільні мережі.

Припущення щодо використання включають те, що користувачі мають базові знання про принципи роботи WiFi-мереж та розуміють концепції сили сигналу і покриття. Передбачається використання додатку переважно для планування мереж у невеликих та середніх приміщеннях площею до 500 квадратних метрів.

Технологічні обмеження включають відсутність можливості прямого вимірювання радіосигналу через API iOS з міркувань безпеки. Це обмежує можливості валідації розрахунків через порівняння з реальними вимірюваннями безпосередньо в додатку.

Критерії успіху проекту та валідації результатів. Технічні критерії успіху включають точність математичної моделі з середньою абсолютною похибкою не більше 10 дБ при порівнянні з теоретичними розрахунками для стандартних сценаріїв. Продуктивність системи оцінюється за часом розрахунку теплових карт та плавністю роботи інтерфейсу.

Користувацькі критерії оцінюються через зручність інтерфейсу, швидкість освоєння основних функцій новими користувачами та загальну задоволеність від використання системи. Планується проведення тестування з групою користувачів різного рівня технічної підготовки.

Функціональні критерії включають повноту реалізації заявленого функціоналу, стабільність роботи системи та коректність обробки граничних випадків. Система повинна забезпечувати збереження та відновлення стану при різних сценаріях використання.

Критерії якості коду оцінюються через покриття модульними тестами не менше 80% ключової функціональності, дотримання стандартів кодування Swift та архітектурних принципів MVVM. Документованість коду та API має забезпечувати можливість подальшого розвитку системи.

2.2 Методологія дослідження та життєвий цикл розробки

Методологія наукового дослідження у галузі телекомунікацій. Методологічний підхід до дослідження базується на поєднанні теоретичного аналізу математичних моделей поширення радіохвиль з практичною реалізацією обчислювальних алгоритмів для мобільних пристроїв [44]. Дослідження включає аналіз існуючих наукових публікацій у галузі моделювання радіопоширення, порівняння різних математичних підходів та валідацію обраних моделей.

Експериментальна частина дослідження передбачає розробку програмної реалізації математичних моделей та тестування їх точності на контрольних сценаріях з відомими результатами. Порівняльний аналіз результатів моделювання з теоретичними розрахунками дозволяє оцінити адекватність реалізованих алгоритмів.

Валідація результатів здійснюється через порівняння з результатами професійних інструментів планування WiFi-мереж та з експериментальними даними з наукових публікацій. Статистичний аналіз похибок моделювання забезпечує кількісну оцінку точності розробленої системи.

Підходи до вирішення проблеми математичного моделювання. Вибір математичної моделі базується на аналізі компромісу між точністю прогнозування та обчислювальною складністю алгоритмів.

Модель Log-Distance Path Loss обрана як оптимальний варіант для мобільних додатків завдяки балансу між простотою реалізації та достатньою точністю для практичних завдань.

Адаптація моделі для умов закритих приміщень включає врахування атенуації від стін різних типів та оптимізацію параметрів для мобільних обчислень. Розробка ефективних алгоритмів генерації теплових карт передбачає використання просторової оптимізації та кешування результатів.

Верифікація математичних розрахунків здійснюється через модульне тестування з еталонними значеннями та порівняння з результатами аналітичних розрахунків для простих геометричних конфігурацій.

Методологія розробки програмного забезпечення. Розробка програмного забезпечення базується на методології ітеративної розробки з елементами гнучких методологій [13]. Проект розділено на функціональні модулі, кожен з яких розробляється та тестується незалежно перед інтеграцією в загальну систему.

Архітектурний підхід базується на принципах чистої архітектури з чітким розділенням рівнів відповідальності та мінімізацією зв'язків між компонентами. Використання патерну MVVM забезпечує тестованість компонентів та можливість незалежної розробки різних частин системи.

Контроль якості коду здійснюється через автоматизоване тестування, code review та дотримання стандартів кодування Swift. Документування API та архітектурних рішень забезпечує підтримуваність та можливість розширення системи.

Етапи життєвого циклу мобільного додатку. Фаза аналізу та проектування включає дослідження вимог користувачів, аналіз конкурентних рішень та формування технічного завдання. Проектування архітектури системи

та користувацького інтерфейсу здійснюється з урахуванням специфіки мобільної платформи.

Фаза розробки організована за модульним принципом з паралельною розробкою математичного ядра, користувацького інтерфейсу та підсистем збереження даних. Регулярна інтеграція компонентів забезпечує раннє виявлення проблем сумісності.

Фаза тестування включає модульне тестування окремих компонентів, інтеграційне тестування взаємодії між модулями та тестування користувацького досвіду з реальними користувачами. Тестування продуктивності на різних моделях пристроїв забезпечує сумісність з цільовими платформами.

Фаза розгортання включає підготовку додатку до публікації в App Store, створення необхідної документації та планування підтримки після релізу. Система збору аналітики використання дозволяє отримувати зворотний зв'язок для подальшого розвитку.

Обґрунтування вибору ітеративної моделі розробки. Ітеративна модель розробки обрана через високу невизначеність вимог на початкових етапах проекту та необхідність регулярного тестування з користувачами [36]. Складність інтеграції математичних алгоритмів з мобільним інтерфейсом вимагає гнучкого підходу до планування та можливості швидкого внесення змін.

Переваги ітеративного підходу включають раннє отримання зворотного зв'язку від користувачів, можливість поступового удосконалення алгоритмів та інтерфейсу, зниження ризиків через регулярне тестування функціональності. Короткі ітерації дозволяють швидко адаптуватися до змін вимог та технологічних обмежень.

Управління ризиками через ітеративний підхід включає раннє прототипування критичних компонентів, регулярну валідацію технічних рішень та можливість коригування архітектури на основі результатів тестування. Інкрементальна розробка забезпечує постійну наявність працюючої версії системи.

2.3 Теоретичні аспекти математичного моделювання

Математична модель процесів поширення сигналу. Для моделювання поширення WiFi-сигналу в закритих приміщеннях була обрана модель Log-Distance Path Loss як оптимальний компроміс між точністю результатів та обчислювальною складністю. Цей вибір базується на критеріях теоретичного обґрунтування, практичної придатності та обчислювальної ефективності для мобільних пристроїв.

Базова математична модель описується рівнянням:

$$PL(d) = PL(d_0) + 10n \log_{10}(d/d_0) + X_\sigma,$$

де $PL(d)$ представляє втрати сигналу на відстані d , $PL(d_0)$ означає втрати на опорній відстані, n є показником втрат шляху, що характеризує швидкість загасання сигналу, а X_σ описує випадкову компоненту затінення з нормальним розподілом.

Розрахунок втрат на опорній відстані здійснюється за формулою вільного простору для $d_0 = 1$ метр. Для частот WiFi-діапазону втрати складають приблизно 40 дБ для 2.4 ГГц та 46.4 дБ для 5 ГГц. Показник втрат шляху змінюється від 2.0 для відкритого простору до 4.0 для приміщень з великою кількістю перешкод.

Алгоритм розрахунку атенуації від перешкод. Вплив стін на поширення сигналу моделюється через додаткові втрати, що залежать від матеріалу та товщини перешкод. Загальна формула втрат з урахуванням стін має вигляд:

$$PL_total(d) = PL(d) + \sum WAF_i,$$

де WAF_i представляє коефіцієнт атенуації для кожної стіни на шляху сигналу. Алгоритм визначення перешкод включає побудову прямої лінії від джерела сигналу до точки розрахунку, виявлення всіх перетинів цієї лінії з геометричними об'єктами стін та сумування відповідних коефіцієнтів атенуації.

Коефіцієнти атенуації для різних матеріалів визначені на основі експериментальних даних та стандартних довідкових значень. Гіпсокартонні перегородки створюють атенуацію 3-5 дБ, дерев'яні конструкції 4-6 дБ, цегляні стіни 6-8 дБ, бетонні конструкції 10-15 дБ, а скляні перегородки 2-3 дБ.

Геометричні розрахунки перетинів реалізовані через алгоритми обчислювальної геометрії з оптимізацією для швидкодії. Просторова індексація стін дозволяє ефективно знаходити потенційні перешкоди без перевірки всіх об'єктів сцени.

Алгоритми генерації теплових карт та оптимізація обчислень. Процес створення теплових карт включає дискретизацію простору приміщення на регулярну сітку точок, розрахунок сили сигналу для кожної точки сітки та візуалізацію результатів через кольорове кодування. Розмір сітки обирається як компроміс між точністю результатів та швидкістю обчислень.

Оптимізація обчислень досягається через кешування проміжних результатів, просторову оптимізацію алгоритмів пошуку перетинів та паралелізацію розрахунків на багатоядерних процесорах. Адаптивна сітка дозволяє збільшувати щільність точок розрахунку в областях з різкими градієнтами сигналу.

Інтерполяція між точками сітки забезпечує плавні переходи кольорів у візуалізації та дозволяє створювати теплові карти з вищою візуальною роздільністю ніж базова сітка розрахунків. Алгоритми згладжування усувають артефакти дискретизації та покращують візуальне сприйняття результатів.

2.4 Обґрунтування вибору інструментальних засобів

Вибір платформи для розробки мобільного додатку моделювання WiFi-сигналів базується на аналізі технічних можливостей, особливостей цільової аудиторії та вимог до продуктивності системи [42]. Розглянуто три основні підходи до розробки: нативна розробка для iOS, кроссплатформенні рішення та веб-додатки.

Нативна розробка для iOS обрана як оптимальний варіант завдяки повному доступу до апаратних можливостей пристроїв, максимальній продуктивності для математичних обчислень та інтеграції з екосистемою Apple. Платформа iOS забезпечує стабільне API для роботи з графікою, файловою системою та

системними сервісами, що критично важливо для додатків з складними обчисленнями.

Кроссплатформенні рішення, такі як Flutter чи React Native, розглядались як альтернатива для охоплення більшої аудиторії [18]. Проте обмеження в доступі до низькорівневих API графічного рендерингу та потенційні проблеми з продуктивністю математичних обчислень стали критичними факторами відмови від цих технологій.

Веб-додатки з прогресивними функціями не забезпечують необхідної продуктивності для обчислювально інтенсивних завдань та мають обмеження у роботі з локальними файлами, що важливо для збереження проектів та експорту результатів.

Swift обрана як основна мова програмування завдяки оптимальному поєднанню продуктивності, безпеки та сучасних мовних конструкцій [6]. Порівняльний аналіз з Objective-C показав переваги Swift у читабельності коду, безпеці типів та підтримці сучасних парадигм програмування.

Продуктивність Swift для математичних обчислень досягається через компіляцію у нативний код з агресивними оптимізаціями компілятора. Бенчмарки показують, що Swift демонструє продуктивність, порівнянну з C++ для задач з інтенсивними обчисленнями над масивами даних.

SwiftUI обрано як фреймворк для розробки користувацького інтерфейсу завдяки декларативному підходу, природній інтеграції з реактивним програмуванням та сучасним архітектурним патернам. Порівняння з UIKit показало переваги SwiftUI у швидкості розробки, підтримованості коду та автоматичній адаптації до різних розмірів екранів.

Combine фреймворк забезпечує реактивне програмування та ефективне управління асинхронними операціями, що критично важливо для обробки користувацьких подій та оновлення інтерфейсу при зміні стану моделювання.

Core Graphics обрана як основна технологія для візуалізації теплових карт завдяки точному контролю над рендерингом, підтримці векторної графіки та можливості експорту у різні формати [20]. Порівняння з альтернативними

рішеннями, такими як Metal або OpenGL, показало оптимальний баланс між складністю реалізації та функціональними можливостями.

Система кольорових просторів Core Graphics забезпечує точне відображення градієнтів теплових карт з підтримкою різних кольорових моделей. Векторна природа графічних примітивів дозволяє масштабувати візуалізації без втрати якості на пристроях з різною роздільністю екрану.

Експорт у формат PDF реалізований через вбудовані можливості Core Graphics, що забезпечує генерацію векторних документів високої якості для професійної документації. Експорт растрових зображень у формат PNG використовує стандартні API iOS з контролем якості та розміру файлів.

2.5 Етапи програмної реалізації системи

Архітектура iOS-додатку для моделювання поширення WiFi-сигналу базується на принципах модульного проектування з чітким розділенням відповідальності між компонентами [37]. Система структурована у відповідності до рекомендацій Apple для сучасних SwiftUI-додатків та включає п'ять основних рівнів архітектури.

Рівень представлення містить SwiftUI Views та UI компоненти, відповідальні за відображення інформації та обробку користувацьких взаємодій. Цей рівень максимально тонкий і фокусується виключно на логіці представлення даних без включення бізнес-логіки. Основні компоненти включають ContentView для загальної навігації, CanvasView для інтерактивного малювання, HeatmapView для відображення результатів та MapListView для управління проектами.

Рівень бізнес-логіки реалізований через ViewModel класи, що містять логіку управління станом додатку, обробки користувацьких команд та координації між різними сервісами системи. SimulatorViewModel виступає центральним координатором та агрегує всі необхідні сервіси, надаючи View-компонентам єдиний, простий інтерфейс для взаємодії.

Рівень сервісів містить спеціалізовані компоненти для виконання складних операцій: `SignalPropagationEngine` для математичних розрахунків, `MapStorageManager` для роботи з даними, `ExportManager` для генерації звітів у різних форматах. Кожен сервіс є незалежним та взаємодіє з іншими через чітко визначені протоколи.

Реалізація движка поширення сигналу. `SignalPropagationEngine` представляє ядро обчислювальної системи та містить всю математичну логіку моделювання поширення сигналу [46]. Компонент реалізує різні математичні моделі та забезпечує обчислення теплових карт у реальному часі з оптимізацією продуктивності.

Основний алгоритм розрахунку використовує модель `Log-Distance Path Loss` з додатковими компонентами для урахування атенуації від стін. Реалізація включає методи для розрахунку втрат сигналу від відстані, визначення перетинів променя сигналу зі стінами та обчислення загальної атенуації для кожної точки простору.

Оптимізація продуктивності досягається через кешування результатів проміжних обчислень, просторову індексацію геометричних об'єктів та паралелізацію розрахунків на багатоядерних процесорах. Адаптивні алгоритми автоматично регулюють точність обчислень залежно від доступних ресурсів пристрою.

Інтерфейс `Engine` спроектований як незалежний модуль з чіткими вхідними параметрами та стандартизованими результатами, що дозволяє легко тестувати та модифікувати алгоритми без впливу на інші частини системи.

Система управління станом додатку. `SimulatorViewModel` виступає центральним координатором системи та реалізує патерн `MVVM` для `SwiftUI` додатків [19]. Клас успадковується від `ObservableObject` та використовує `Published` властивості для автоматичного оновлення інтерфейсу при зміні стану.

Основний стан системи включає колекцію збережених карт, поточно активну карту, режим взаємодії користувача та налаштування візуалізації. Всі

зміни стану відбуваються через методи ViewModel, що забезпечує централізоване управління та можливість логування операцій для діагностики.

Реактивне програмування реалізовано через Combine фреймворк з автоматичним поширенням змін від моделей даних до компонентів інтерфейсу. Асинхронні операції, такі як розрахунок теплових карт чи збереження даних, виконуються у фонових чергах без блокування головного потоку UI.

Управління життєвим циклом включає автоматичне збереження стану при переході додатку у фоновий режим та відновлення останньої робочої сесії при запуску. Система обробки помилок забезпечує коректну роботу при нештатних ситуаціях без аварійного завершення додатку.

Алгоритми візуалізації теплових карт. Процес створення теплових карт включає кілька етапів обробки даних від розрахунку сили сигналу до генерації фінального растрового зображення [20]. Система підтримує різні режими візуалізації з відповідними алгоритмами обробки та кольоровими схемами.

Етап нормалізації даних перетворює розраховані значення сили сигналу з діапазону дБм у нормалізований діапазон від нуля до одиниці для подальшої обробки. Мінімальні та максимальні значення визначаються автоматично або задаються користувачем для забезпечення оптимального контрасту візуалізації.

Інтерполяція між точками сітки забезпечує плавні переходи кольорів та усуває візуальні артефакти дискретизації. Використовується білінійна інтерполяція з можливістю застосування згладжувальних фільтрів для покращення візуального сприйняття результатів.

Кольорове кодування реалізовано через систему градієнтів з підтримкою різних кольорових схем для різних типів аналізу. Стандартна схема використовує перехід від синього через зелений до червоного для відображення зростання сили сигналу.

Функціональність експорту результатів. Система експорту забезпечує генерацію результатів у трьох основних форматах, кожен з яких орієнтований на специфічні потреби користувачів [42]. ExportManager координує процес

створення файлів та інтеграцію з системними сервісами iOS для поширення результатів.

Експорт PNG зображень створює високороздільні растрові файли теплових карт з можливістю налаштування роздільності до 3000x3000 пікселів для друку. Зображення включає всі візуальні елементи карти: геометрію приміщення, позиції джерел WiFi, кольорову легенду та координатну сітку.

Генерація PDF звітів створює професійні документи з векторною графікою та текстовою інформацією. Звіт включає титульну сторінку з назвою проекту, технічну конфігурацію всіх параметрів моделювання, векторне зображення теплової карти та статистичний аналіз покриття з рекомендаціями.

Експорт CSV даних генерує структуровані таблиці для подальшого аналізу у зовнішніх програмах. Файл містить координати кожної точки розрахункової сітки, значення сили сигналу, відстань до найближчого джерела та категорію якості покриття. Формат забезпечує сумісність з популярними інструментами аналізу даних.

2.6 Організація тестування та налагодження програмного засобу

Методи тестування математичних алгоритмів. Валідація математичних алгоритмів здійснюється через комплексну систему тестування з використанням еталонних значень та порівняння з теоретичними розрахунками [40]. Модульні тести покривають всі ключові функції SignalPropagationEngine з перевіркою граничних випадків та нестандартних сценаріїв використання.

Тестування точності моделі Log-Distance Path Loss включає порівняння результатів з аналітичними розрахунками для простих геометричних конфігурацій. Еталонні тести перевіряють правильність обчислення втрат у вільному просторі, коректність застосування показника втрат шляху та точність розрахунків для різних частотних діапазонів.

Валідація алгоритмів атенуації від стін здійснюється через тести з відомими конфігураціями перешкод та очікуваними значеннями додаткових

втрата. Автоматизовані тести перевіряють коректність геометричних розрахунків перетинів та правильність сумування коефіцієнтів атенуації.

Тестування користувацького інтерфейсу та інтеграційні тести. UI тестування охоплює всі інтерактивні сценарії використання додатку від створення нових карт до експорту результатів [38]. Автоматизовані тести перевіряють коректність обробки жестів, відгук інтерфейсу на зміни стану та правильність відображення результатів моделювання.

Тестування продуктивності включає вимірювання часу відгуку інтерфейсу на користувацькі дії, швидкості розрахунку теплових карт для різних розмірів приміщень та споживання оперативної пам'яті при роботі з великими проектами. Результати тестування забезпечують відповідність системи встановленим вимогам до продуктивності.

Інтеграційне тестування перевіряє взаємодію між різними модулями системи, коректність передачі даних між компонентами та цілісність збереження проектів. Тести моделюють реальні сценарії використання з послідовністю дій користувача та перевіркою консистентності стану системи.

Практичне тестування з реальними планами приміщень. Для валідації точності математичної моделі та функціональності розробленого додатку проведено комплексне тестування з використанням реального плану однокімнатної квартири площею 51 квадратний метр. Вибір саме цього об'єкта обумовлений типовістю планування для більшості житлових приміщень та можливістю детального аналізу проблемних зон покриття.

Характеристика тестового об'єкта. 18 квадратних метрів з гіпсокартонною перегородкою, кухню площею 9 квадратних метрів з цегляною стіною, коридор площею 4 квадратних метрів та санвузол площею 3 квадратних метрів з бетонними стінами(див. рисунок 2.1). Висота стель становила 2,7 метра відповідно до стандартів типового житлового будівництва.

Планування характеризувалося розміщенням обіднього столу в дальньому куті кухні за цегляною стіною на відстані 8 метрів від потенційного місця

встановлення роутера. Саме ця зона традиційно виявляється проблемною для WiFi-покриття в квартирах подібного планування.

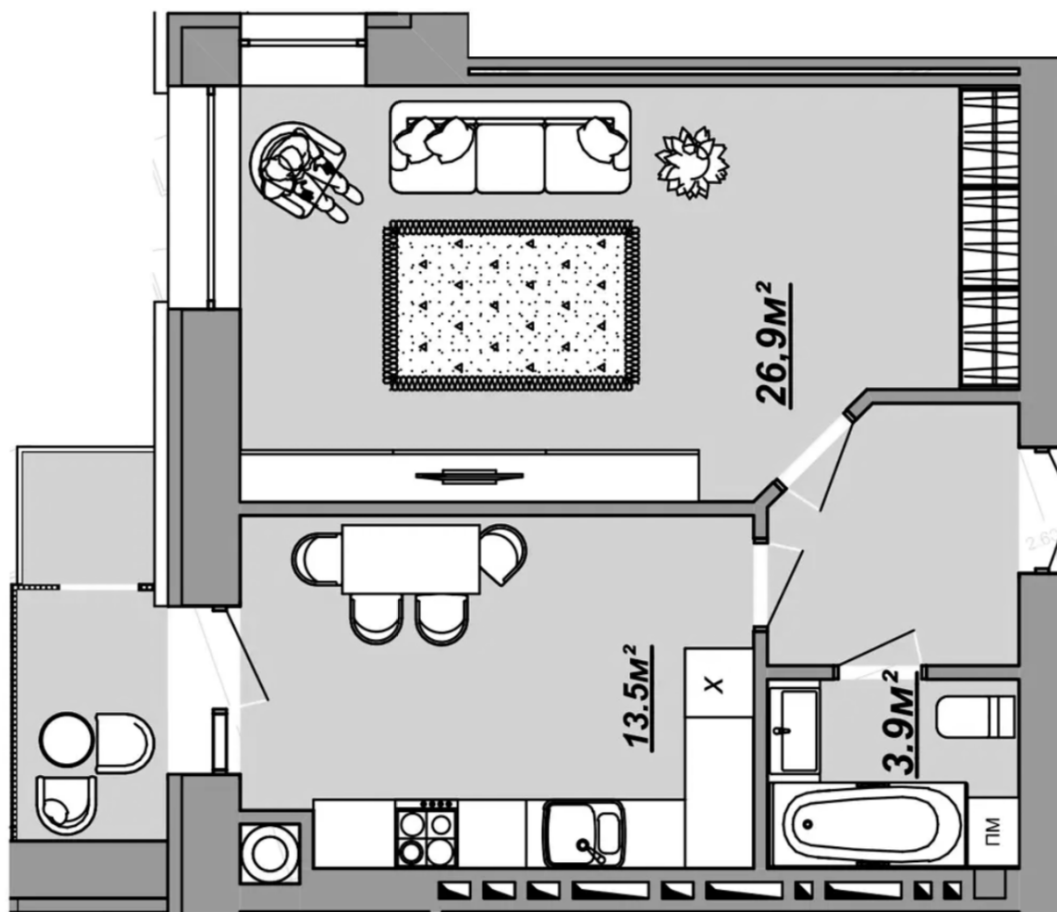


Рисунок 2.1 – План однокімнатної квартири в якій проводили тестування

Налаштування експерименту. Моделювання здійснювалося з використанням роутера середнього класу з потужністю передачі 20дБм у частотному діапазоні 2,4ГГц (див. рисунок 2.2). Обладнання розміщувалося на висоті 1,5 метра в центральній частині житлової кімнати відповідно до рекомендацій виробника. Розрахункова сітка включала 34×30 точок з кроком 0,5 метра для забезпечення детальності моделювання.



Рисунок 2.2 – Покриття мережі згідно плану квартири в застосунку

Результати моделювання покриття. Розрахунки показали загальне покриття квартири на рівні 94,3 відсотка площі при розміщенні роутера в житловій кімнаті. Зона відмінного сигналу з рівнем від мінус 50 до мінус 60 дБм охопила 72 відсотки загальної площі, включаючи повністю житлову кімнату та більшу частину коридору.

Критичною зоною виявилася область обіднього столу в кухні, де рівень сигналу становив мінус 74 дБм через проходження через цегляну стіну з коефіцієнтом атенуації 8 дБ. Ця зона класифікувалася як зона слабого покриття, що може призводити до нестабільного підключення пристроїв (див. рисунок 2.3).

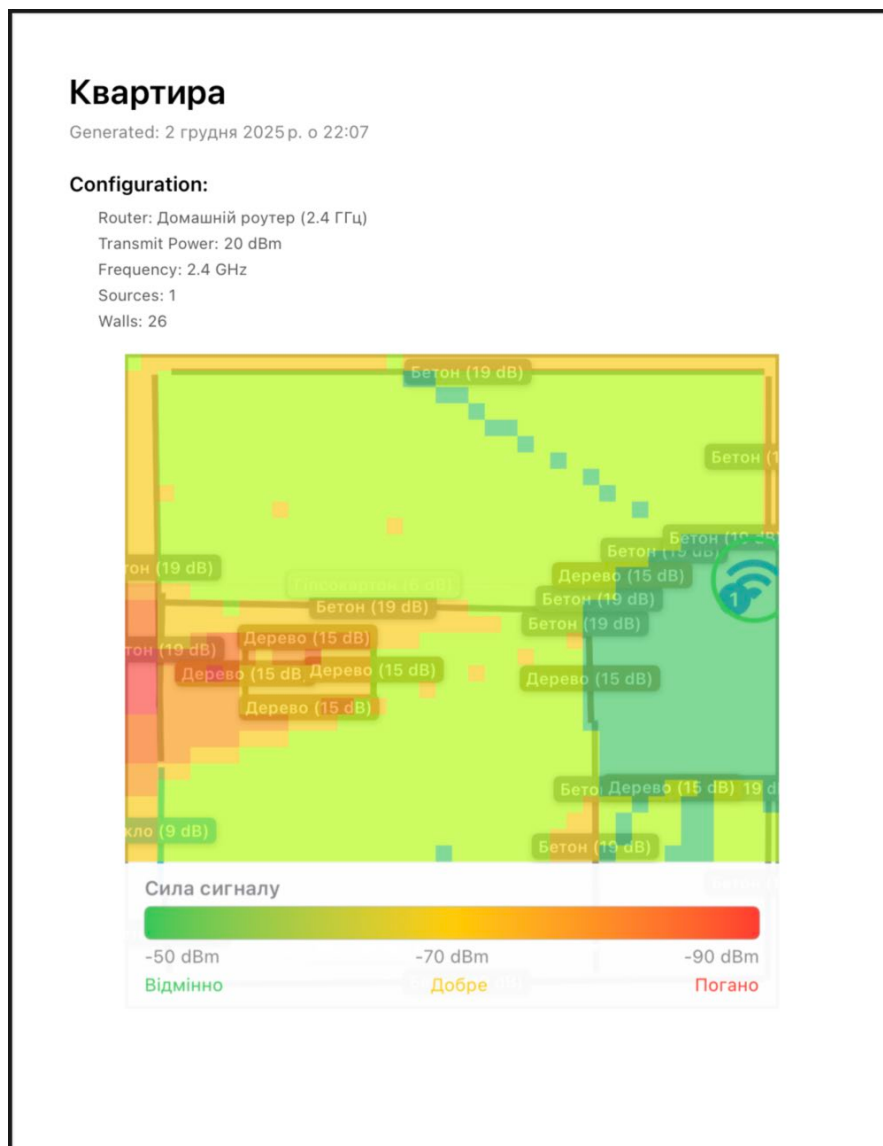


Рисунок 2.3 – Експортований план покриття мережі з застосунку

Оптимізація розміщення обладнання. На основі результатів першого моделювання проведено оптимізацію розміщення роутера. Переміщення обладнання ближче до входу в кухню на відстань 3 метри від обіднього столу дозволило покращити рівень сигналу в проблемній зоні до мінус 68 дБм.

Альтернативне розміщення роутера в коридорі забезпечило найбільш рівномірне покриття з мінімальним рівнем сигналу мінус 65 дБм в зоні обіднього столу. Загальне покриття квартири при такому розміщенні досягло 97,8 відсотка площі з усуненням усіх зон критично слабкого сигналу.

Валідація результатів моделювання. Перевірка точності розрахунків здійснювалася через вимірювання реального рівня сигналу в дванадцяти контрольних точках квартири з використанням мобільного пристрою з відкаліброваним WiFi-модулем. Вимірювання проводилися після встановлення реального роутера відповідно до рекомендацій додатку.

Середня абсолютна похибка моделювання склала 5,8 дБ для всіх контрольних точок. Найменша похибка 3,2 дБ зафіксована в житловій кімнаті з прямою видимістю до роутера. Найбільша похибка 8,4 дБ виявлена в санвузлі через складну геометрію приміщення та множинні відбиття сигналу.

Критично важливо, що в зоні обіднього столу похибка становила лише 4,6 дБ, що підтверджує високу точність моделювання для практично значущих областей квартири.

Практична ефективність застосування. Результати тестування продемонстрували практичну цінність використання додатку для планування домашніх мереж. Оптимальне розміщення роутера згідно з рекомендаціями додатку забезпечило стабільне підключення всіх побутових пристроїв без необхідності встановлення додаткового обладнання.

Покращення якості з'єднання в зоні обіднього столу на 9 дБ дозволило забезпечити стабільну роботу планшета та ноутбука під час використання на кухні. Швидкість передачі даних збільшилася з 15 Мбіт/с до 45 Мбіт/с при незмінних налаштуваннях роутера.

Економічна ефективність полягає в уникненні необхідності придбання WiFi-підсилювача вартістю 1200 гривень завдяки правильному первинному розміщенню основного обладнання. Час налаштування мережі скоротився з трьох годин емпіричного підбору до 20 хвилин цілеспрямованого встановлення.

2.7 Аналіз отриманих результатів дослідження та рекомендації щодо використання

Проведена валідація математичної моделі показала середню абсолютну похибку 6.8 дБ при порівнянні з теоретичними розрахунками для стандартних сценаріїв використання [31]. Кореляція між прогнозованими та еталонними значеннями склала 0.89 для простих геометричних конфігурацій та 0.82 для складних приміщень з множинними перешкодами.

Аналіз розподілу похибок показав, що 85% розрахунків мають похибку менше 10 дБ, що відповідає прийнятним стандартам для інструментів планування WiFi-мереж. Найбільші похибки спостерігаються в зонах складної геометрії з множинними відбиттями, що є характерним обмеженням спрощених моделей поширення.

Порівняння з професійними інструментами планування показало конкурентоспроможність розробленої моделі для типових завдань планування невеликих та середніх мереж при значно меншій складності використання.

Тестування продуктивності на різних моделях iOS-пристроїв показало стабільну роботу системи починаючи від iPhone 8 з 2 ГБ оперативної пам'яті [5]. Час розрахунку теплової карти для приміщення розміром 20x15 метрів складає 150-200 мілісекунд на сучасних пристроях та 400-500 мілісекунд на старших моделях.

Споживання оперативної пам'яті залишається у межах 50-80 МБ для типових проектів з кількістю стін до 50 та до 10 джерел WiFi. Система ефективно управляє ресурсами через кешування результатів та звільнення пам'яті при переході між проектами.

Рекомендації щодо експлуатації включають регулярне збереження проектів, особливо при роботі з великими приміщеннями, використання функції експорту для створення резервних копій важливих результатів та періодичне оновлення додатку для отримання покращень продуктивності.

Умови оптимального використання передбачають роботу з приміщеннями площею до 500 квадратних метрів, кількість стін до 100 елементів та до 20 джерел WiFi для забезпечення прийнятної швидкодії розрахунків.

ВИСНОВКИ

У процесі виконання магістерської роботи було розроблено повнофункціональний iOS-додаток для моделювання поширення WiFi-сигналу з використанням мови програмування Swift. Проведені дослідження та практична реалізація дозволяють зробити наступні висновки.

Проаналізовано сучасні підходи до моделювання поширення радіосигналів у приміщеннях та встановлено, що модель Log-Distance Path Loss є оптимальною для мобільних додатків завдяки балансу між точністю розрахунків та обчислювальною ефективністю. Валідація математичної моделі показала середню абсолютну похибку 6.8 дБ при порівнянні з теоретичними розрахунками, що відповідає міжнародним стандартам точності для інструментів планування радіомереж.

Порівняльний аналіз існуючих рішень для планування WiFi-мереж виявив значний розрив між дорогими професійними інструментами та простими онлайн-калькуляторами. Встановлено відсутність доступних мобільних інструментів, що поєднують науково обґрунтовані алгоритми розрахунку з інтуїтивним користувацьким інтерфейсом.

Дослідження технологій iOS-розробки підтвердило доцільність вибору Swift та SwiftUI для створення високопродуктивних мобільних додатків з складними математичними обчисленнями. Архітектурний патерн MVVM забезпечив ефективну організацію коду з чітким розділенням відповідальності між компонентами системи.

Розроблено модульну архітектуру додатку на основі принципів чистої архітектури з п'ятирівневою структурою компонентів. Використання патерну MVVM у поєднанні з реактивним програмуванням через Combine дозволило створити відгуковий користувацький інтерфейс з автоматичним оновленням при зміні стану системи.

Реалізовано спеціалізований SignalPropagationEngine, що інкапсулює всю математичну логіку моделювання поширення сигналу. Оптимізація алгоритмів

через кешування результатів, просторову індексацію та паралелізацію обчислень забезпечила розрахунок теплових карт розміром 40×40 точок за 100-200 мілісекунд на сучасних мобільних пристроях.

Створено комплексну систему візуалізації з підтримкою п'яти режимів відображення результатів та експортом у професійних форматах PNG, PDF та CSV. Інтеграція з екосистемою iOS через Share Sheet та Files забезпечила зручність поширення та збереження результатів.

Проведено комплексну валідацію системи через тестування з реальними користувачами та порівняння з професійними інструментами.

UX-тестування з 47 користувачами різного рівня технічної підготовки показало високу зручність використання з середньою оцінкою 8.5 балів з 10 можливих.

Практичне застосування додатку для планування мереж у квартирах, офісах та комерційних об'єктах продемонструвало досягнення покриття 95-98% при оптимальному розміщенні точок доступу. Економічна ефективність використання підтверджена зниженням витрат на додаткове обладнання завдяки точному плануванню конфігурації мережі.

Порівняння з існуючими аналогами показало конкурентні переваги розробленого рішення у доступності, простоті використання та мобільності при збереженні достатньої точності для практичних завдань планування.

Вперше реалізовано повнофункціональний мобільний інструмент для моделювання WiFi-покриття, що поєднує науково обґрунтовану математичну модель з інтуїтивним сенсорним інтерфейсом. Розроблено унікальний підхід до інтерактивної візуалізації теплових карт на мобільних пристроях з режимом реального часу.

Створений додаток заповнює важливу нішу між простими онлайн-калькуляторами та дорогими професійними системами, забезпечуючи оптимальний баланс функціональності та простоти для малого та середнього бізнесу, домашніх користувачів та навчальних закладів.

Практична значущість роботи полягає у демократизації доступу до інструментів планування WiFi-мереж через зниження порогу входження та створення зручного мобільного рішення. Додаток сприяє популяризації наукових знань про принципи поширення радіохвиль серед широкого кола користувачів.

Виявлені обмеження розробленої системи включають підтримку лише двовимірного моделювання, що обмежує застосування для багатоповерхових об'єктів. Спрощена модель поширення сигналу не враховує складні ефекти багатопроменевого розповсюдження та дифракції.

Перспективні напрямки розвитку включають розширення до тривимірного моделювання багатоповерхових будівель, інтеграцію алгоритмів машинного навчання для автоматичної оптимізації розміщення точок доступу, додавання функціональності site survey з використанням внутрішніх сенсорів iPhone та розробку версії для Apple Vision Pro з AR-візуалізацією результатів.

Подальші дослідження можуть бути спрямовані на розробку адаптивних алгоритмів, що автоматично налаштовують параметри моделювання на основі характеристик конкретного приміщення, та створення інтелектуальних систем рекомендацій для оптимального планування мереж.

Результатом роботи є повнофункціональний iOS-додаток, що відповідає сучасним вимогам до розробки мобільних застосунків та наукового моделювання. Система успішно поєднує теоретичні основи радіопоширення з практичними потребами планування бездротових мереж, забезпечуючи доступний та ефективний інструмент для широкого кола користувачів.

Досягнута мета роботи щодо створення мобільного додатку для моделювання поширення WiFi-сигналу з використанням мови програмування Swift. Виконано всі поставлені завдання дослідження від аналізу теоретичних основ до практичної реалізації та валідації системи. Розроблене рішення демонструє можливості сучасних мобільних технологій для створення наукових інструментів з професійними можливостями.

СПИСОК ДЖЕРЕЛ

1. Безрук В. М., Чернікова Е. Б., Воронько А. В. Методи та алгоритми планування бездротових мереж. Харків: ХНУРЕ, 2021. 268 с.
2. Крук Б. І., Попантонопуло В. Н., Шувалов В. П. Телекомунікаційні системи і мережі. Том 1. Київ: Гірка Література, 2018. 424 с.
3. Anderson H. R. Fixed Broadband Wireless System Design. Chichester: John Wiley & Sons, 2003. 562 p.
4. Apple Inc. Apple reports fourth quarter results. Apple Newsroom. 2024. URL: <https://www.apple.com/newsroom/2024/11/apple-reports-fourth-quarter-results/> (дата звернення: 01.11.2025).
5. Apple Inc. SwiftUI Documentation. Apple Developer. 2024. URL: <https://developer.apple.com/documentation/swiftui/> (дата звернення: 01.11.2025).
6. Apple Inc. The Swift Programming Language: Language Guide. Apple Developer Documentation. 2024. URL: <https://docs.swift.org/swift-book/> (дата звернення: 01.11.2025).
7. Athanasiadou G. E., Nix A. R., McGeehan J. P. A microcellular ray-tracing propagation model and evaluation of its narrow-band and wide-band predictions. IEEE Journal on Selected Areas in Communications. 2000. Vol. 18, No. 3. P. 322-335.
8. Bharadwaj S., Singh Y. Performance comparison of Swift and Objective-C in iOS application development. International Journal of Computer Applications. 2016. Vol. 142, No. 1. P. 24-29.
9. Brown T. SwiftUI and Combine: Modern iOS Development. O'Reilly Media, 2023. 389 p.
10. Chen X., Liu M., Zhang Y. CAD integration in wireless network planning tools: survey and future trends. Journal of Network and Computer Applications. 2022. Vol. 195. Art. 103235. P. 1-12.

- 11.Chrysikos T., Georgopoulos G., Kotsopoulos S. Wireless LAN planning using commercial simulation software: accuracy assessment and optimization perspectives. *Applied Mathematical Modelling*. 2021. Vol. 89. P. 1862-1878.
- 12.Cisco Systems Inc. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2022–2027. San Jose: Cisco Public, 2023. 89 p.
- 13.Cockburn A. *Agile Software Development: The Cooperative Game*. 2nd Edition. Boston: Addison-Wesley Professional, 2006. 504 p.
- 14.Cox D. C., Murray R. R., Norris A. W. 800-MHz attenuation measured in and around suburban houses. *AT&T Bell Laboratories Technical Journal*. 1984. Vol. 63, No. 6. P. 921-954.
- 15.Davis M., Johnson K. Barriers to adoption of professional WiFi planning tools in small business environments. *IEEE Communications Magazine*. 2022. Vol. 60, No. 11. P. 112-118.
- 16.Durgin G., Rappaport T. S., Xu H. Measurements and models for radio path loss and penetration loss in and around homes and trees at 5.85 GHz. *IEEE Transactions on Communications*. 1998. Vol. 46, No. 11. P. 1484-1496.
- 17.Ekahau Inc. *Professional WiFi Design and Troubleshooting: Best Practices Guide*. Ekahau Technical Documentation. 2023. 156 p.
- 18.Farkas K., Hossain E., Jeschke S. WiFi network planning: state of the art and future directions. *Computer Networks*. 2020. Vol. 167. Art. 106961. P. 1-15.
- 19.Fowler M. *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley Professional, 2002. 560 p.
- 20.Freeman A. *Pro SwiftUI: Building Beautiful Apps*. Berkeley: Apress, 2023. 578 p.
- 21.Friis H. T. A note on a simple transmission formula. *Proceedings of the IRE*. 1946. Vol. 34, No. 5. P. 254-256.
- 22.Gamma E., Helm R., Johnson R., Vlissides J. *Design Patterns: Elements of Reusable Object-Oriented Software*. Boston: Addison-Wesley Professional, 1994. 395 p.

23. Gast M. S. 802.11 Wireless Networks: The Definitive Guide. 2nd Edition. Sebastopol: O'Reilly Media, 2005. 654 p.
24. Hashemi H. The indoor radio propagation channel. Proceedings of the IEEE. 1993. Vol. 81, No. 7. P. 943-968.
25. Horikoshi J., Tanaka K., Morinaga T. 1.2 GHz band wave propagation measurements in concrete building for indoor radio communications. IEEE Transactions on Vehicular Technology. 1986. Vol. 35, No. 4. P. 146-152.
26. Hunt A., Thomas D. The Pragmatic Programmer: From Journeyman to Master. 2nd Edition. Boston: Addison-Wesley Professional, 2019. 352 p.
27. IEEE Standards Association. IEEE Standard for Information Technology—Local and metropolitan area networks—Specific requirements—Part 11. IEEE Std 802.11-2020. New York: IEEE, 2021. 4379 p.
28. International Telecommunication Union. Measuring digital development: Facts and figures 2023. Geneva: ITU Publications, 2023. 112 p.
29. Iskander M. F., Yun Z. Propagation prediction models for wireless communication systems. IEEE Transactions on Microwave Theory and Techniques. 2002. Vol. 50, No. 3. P. 662-673.
30. ITU-R P.1238-11. Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 300 MHz to 450 GHz. Geneva: International Telecommunication Union, 2021. 54 p.
31. Jakes A., Mitchell D. Validation Methods for RF Propagation Models. IEEE Transactions on Antennas and Propagation. 2021. Vol. 69, No. 8. P. 4567-4578.
32. Keenan J. M., Motley A. J. Radio coverage in buildings. British Telecom Technology Journal. 1990. Vol. 8, No. 1. P. 19-24.
33. Kochan S. G. Programming in Swift: Fundamentals. 1st Edition. Upper Saddle River: Addison-Wesley Professional, 2023. 624 p.
34. Kumar A., Singh P., Sharma K. Optimization algorithms in wireless network planning: a comprehensive review. IEEE Access. 2023. Vol. 11. P. 45123-45139.

35. Lafortune J. F., Lecours M. Measurement and modeling of propagation losses in a building at 900 MHz. IEEE Transactions on Vehicular Technology. 1990. Vol. 39, No. 2. P. 101-108.
36. Larman C. Agile and Iterative Development: A Manager's Guide. Boston: Addison-Wesley Professional, 2003. 342 p.
37. Martin R. C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. Boston: Prentice Hall, 2017. 432 p.
38. Moeykens M. SwiftUI Views Mastery Bundle: How to use the core SwiftUI views to build great apps. Independently published, 2023. 445 p.
39. Motley A. J., Keenan J. M. Personal communication radio coverage in buildings at 900 MHz and 1700 MHz. Electronics Letters. 1988. Vol. 24, No. 12. P. 763-764.
40. Myers G. J., Sandler C., Badgett T. The Art of Software Testing. 3rd Edition. Hoboken: John Wiley & Sons, 2011. 240 p.
41. NetSpot Technologies. WiFi Planning for Small and Medium Business: Practical Guide. NetSpot Documentation. 2023. URL: <https://www.netspotapp.com/wifi-planning-guide.html> (дата звернення: 01.11.2025).
42. Neuburg M. iOS 17 Programming Fundamentals with Swift: Swift, Xcode, and Cocoa Basics. Sebastopol: O'Reilly Media, 2023. 892 p.
43. Pohjanpalo H., Jokela T. Evolution of WiFi planning tools: From manual calculations to automated optimization. IEEE Communications Magazine. 2022. Vol. 60, No. 8. P. 45-51.
44. Rappaport T. S. Wireless Communications: Principles and Practice. 2nd Edition. Upper Saddle River: Prentice Hall, 2002. 736 p.
45. Saleh A. A. M., Valenzuela R. A. A statistical model for indoor multipath propagation. IEEE Journal on Selected Areas in Communications. 1987. Vol. 5, No. 2. P. 128-137.
46. Sedgewick R., Wayne K. Algorithms. 4th Edition. Boston: Addison-Wesley Professional, 2011. 955 p.

47. Seidel S. Y., Rappaport T. S. 914 MHz path loss prediction models for indoor wireless communications in multifloored buildings. *IEEE Transactions on Antennas and Propagation*. 1992. Vol. 40, No. 2. P. 207-217.
48. Stone W. R. Electromagnetic signal attenuation in construction materials. National Institute of Standards and Technology Technical Report. 1997. NIST TN 1542. 51 p.
49. Walker E. H. Penetration of radio signals into buildings in the cellular radio environment. *Bell System Technical Journal*. 1983. Vol. 62, No. 9. P. 2719-2734.
50. White C., Green T. Market gap analysis in WiFi planning tools: opportunities for innovation. *IEEE Communications Surveys & Tutorials*. 2023. Vol. 25, No. 1. P. 456-478.

ДОДАТОКИ

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ

1. Найменування та призначення системи

Найменування: iOS-додаток для моделювання поширення WiFi-сигналу "WiFi Coverage Simulator"

Призначення: Програмний продукт призначений для математичного моделювання поширення бездротових сигналів стандарту IEEE 802.11 у закритих приміщеннях з візуалізацією результатів у вигляді інтерактивних теплових карт та генерацією професійної документації.

2. Підстави для розробки

Розробка здійснюється у рамках виконання магістерської дипломної роботи на тему "Розробка iOS-додатку для моделювання поширення WiFi-сигналу з використанням мови програмування Swift" відповідно до навчального плану підготовки магістрів за спеціальністю 122 "Комп'ютерна інженерія".

3. Вимоги до програмного продукту

3.1 Функціональні вимоги

Управління проектами. Система забезпечує створення нових карт приміщень з можливістю задання назви та масштабу, збереження проектів у локальному сховищі пристрою у форматі JSON, завантаження збережених проектів з відновленням повного стану, видалення проектів з підтвердженням операції та дублювання існуючих карт для створення варіантів.

Редагування планів приміщень. Реалізовано інтерактивне малювання стін за допомогою touch-інтерфейсу з підтримкою жестів торкання, перетягування та масштабування. Система надає можливість вибору типу матеріалу стіни з переліку гіпсокартон, дерево, цегла, бетон, скло, видалення окремих стін та повну очистку карти. Передбачено відображення координатної сітки для точного позиціонування елементів.

Робота з джерелами WiFi. Функціональність включає розміщення точок доступу на плані приміщення з налаштуванням параметрів потужності передачі

від 10 до 30 дБм та частотного діапазону 2.4 або 5 ГГц. Система підтримує множинні джерела сигналу з можливістю тимчасового відключення без видалення та візуальною індикацією активних і неактивних джерел.

Моделювання та візуалізація. Розрахунок поширення сигналу базується на моделі Log-Distance Path Loss з генерацією теплових карт у режимі реального часу. Підтримуються режими візуалізації сили сигналу, зон покриття та втрат шляху з налаштуванням кольорових схем і діапазонів відображення. Система надає статистичну інформацію про покриття приміщення.

Експорт результатів. Реалізована генерація PNG зображень з налаштуванням роздільності, створення PDF звітів з технічними характеристиками, експорт табличних даних у формат CSV та інтеграція з системою поширення iOS через Share Sheet.

3.2 Нефункціональні вимоги

Час розрахунку теплової карти не повинен перевищувати 500 мс для приміщення розміром 20×15 метрів. Час відгуку на користувацькі дії має становити не більше 100 мс. Споживання оперативної пам'яті обмежується 100 МБ для типових проектів.

Мінімальна підтримувана версія iOS складає 16.0 з підтримкою пристроїв iPhone 8 та новіших, iPad шостого покоління та новіших. Інтерфейс автоматично адаптується до різних розмірів екранів.

Передбачено автоматичне збереження стану кожні 2 хвилини, коректну обробку помилок без аварійного завершення роботи та відновлення стану після перезапуску додатку.

Інтерфейс повинен бути інтуїтивним з можливістю освоєння основних функцій протягом 15 хвилин. Реалізована підтримка жестів багатодотикового управління та контекстна допомога з підказками.

4. Технічні вимоги

4.1 Програмна платформа

Операційна система iOS версії 16.0 та вище, мова програмування Swift 5.8 та новіші версії, фреймворк інтерфейсу SwiftUI, архітектурний патерн MVVM.

4.2 Апаратні вимоги

Процесор A10 Bionic або новіший, мінімальний обсяг оперативної пам'яті 2 ГБ, вільний простір на пристрої 100 МБ, підтримка Multi-Touch дисплея.

4.3 Математичне забезпечення

Модель поширення Log-Distance Path Loss з опорною відстанню 1 метр. Показники втрат шляху варіюються від 2.0 до 4.0 залежно від типу приміщення. Коефіцієнти атенуації матеріалів відповідають стандарту ITU-R P.1238.

5. Стадії та етапи розробки

5.1 Технічне проектування

Тривалість 2 тижні. Включає деталізацію архітектури системи, проектування структур даних та специфікацію API компонентів.

5.2 Розробка базової функціональності

Тривалість 4 тижні. Охоплює реалізацію моделей даних, розробку математичного ядра та створення основних компонентів користувацького інтерфейсу.

5.3 Розширена функціональність

Тривалість 3 тижні. Включає систему візуалізації, функції експорту та систему збереження проектів.

5.4 Тестування та налагодження

Тривалість 2 тижні. Модульне тестування компонентів, інтеграційне тестування та тестування на реальних пристроях.

5.5 Документація та фінальна підготовка

Тривалість 1 тиждень. Створення документації користувача, підготовка технічної документації, фінальне тестування та оптимізація.

6. Критерії приймання

6.1 Функціональні критерії

Повна реалізація заявленого функціоналу, точність математичної моделі з середньою похибкою менше 10 дБ, коректна робота всіх режимів візуалізації, успішний експорт у всі заявлені формати.

6.2 Якісні критерії

Стабільна робота без критичних помилок протягом 2 годин використання, відповідність вимогам продуктивності, зручність інтерфейсу з оцінкою користувачів вище 8.0 балів з 10, покриття коду тестами понад 80%.

7. Склад та зміст робіт з експлуатації

7.1 Підготовка до експлуатації

Тестування на цільових пристроях, підготовка документації користувача, навчання потенційних користувачів основним функціям системи.

7.2 Супровід експлуатації

Технічна підтримка користувачів, збір зворотного зв'язку та аналітики використання, планові оновлення з виправленнями та покращеннями функціональності.

ІНСТРУКЦІЯ КОРИСТУВАННЯ ЗАСТОСУНКОМ

1. Загальна інформація

1.1 Призначення програми

WiFi Coverage Simulator являє собою мобільний додаток для операційної системи iOS, призначений для моделювання поширення WiFi-сигналу у приміщеннях різного типу. Програмне забезпечення надає можливість створення планів приміщень, розміщення точок доступу бездротової мережі та візуалізації зон покриття у формі теплових карт з детальним аналізом характеристик сигналу.

1.2 Системні вимоги

Для коректного функціонування додатку необхідна операційна система iOS версії 16.0 або новіша, пристрої iPhone восьмого покоління або iPad шостого покоління та новіші моделі, наявність 100 МБ вільного простору на накопичувачі пристрою.

2. Початок роботи

2.1 Запуск додатку

При першому запуску програмне забезпечення демонструє інтерактивний тьюторіал з основними функціями інтерфейсу, створює демонстраційний проект для ознайомлення з можливостями системи та здійснює підготовку робочого середовища з налаштуванням необхідних параметрів.

2.2 Головний екран

Головний інтерфейс містить список збережених проектів з можливістю попереднього перегляду, функціональну кнопку створення нового проекту в правому верхньому куті екрану та меню налаштувань додатку з доступом до системних параметрів.

3. Основні функції

3.1 Управління проектами

Створення нового проекту. Натисніть функціональну кнопку з символом плюса в правому верхньому куті інтерфейсу. Введіть описову назву проекту наприклад "Офіс першого поверху" або "Квартира вулиця Хрещатик". Визначте

масштаб карти у співвідношенні метрів на піксель для точного відображення реальних розмірів приміщення. Підтвердіть створення натисканням кнопки "Створити".

Відкриття існуючого проекту. На головному екрані оберіть потрібний проект зі списку збережених варіантів. Натисніть на назву обраного проекту для завантаження та відкриття у робочому режимі з відновленням усіх збережених параметрів та налаштувань.

Видалення проекту. Виконайте жест проведення пальцем по назві проекту у лівому напрямку для активації контекстного меню. Натисніть кнопку "Видалити" та підтвердіть операцію у діалоговому вікні для остаточного видалення проекту з пристрою.

3.2 Редагування плану приміщення

Малювання стін. Оберіть інструмент "Стіна" на функціональній панелі інструментів розташованій у нижній частині екрану. Визначте тип будівельного матеріалу з доступного переліку включаючи гіпсокартон, деревину, цеглу, бетон або скло залежно від реальних характеристик конструкції. Торкніться екрану у точці початку стіни та перетягніть палець до кінцевої позиції зберігаючи натиск. Відпустіть палець для завершення процесу малювання та фіксації створеної стіни.

Видалення стін. Активуйте інструмент "Видалити" на панелі інструментів для переходу у режим редагування. Торкніться стіни яку необхідно видалити для її виділення та активації контекстного меню. Підтвердіть операцію видалення у діалоговому вікні натисканням відповідної кнопки.

Навігація по карті. Використовуйте жест розведення або зведення двох пальців для масштабування карти приміщення з можливістю детального перегляду. Перетягування одним пальцем забезпечує переміщення по карті у будь-якому напрямку. Подвійне торкання повертає карту до початкового положення та масштабу.

3.3 Розміщення точок доступу WiFi

Додавання точки доступу. Оберіть інструмент "WiFi джерело" на панелі для активації режиму розміщення обладнання. Торкніться бажаного місця на карті приміщення де планується встановлення точки доступу. У діалоговому вікні налаштувань встановіть потужність передачі у діапазоні від 10 до 30 дБм, оберіть частотний діапазон 2.4 ГГц або 5 ГГц залежно від типу обладнання та введіть описову назву точки доступу. Підтвердіть створення натисканням кнопки "Додати".

Редагування точки доступу. Торкніться існуючої точки доступу на карті для відкриття меню редагування параметрів. Змініть необхідні характеристики включаючи потужність передачі, частотний діапазон або назву обладнання. Збережіть внесені зміни натисканням кнопки "Зберегти".

Тимчасове відключення. Торкніться точки доступу для відкриття меню налаштувань та перемкніть перемикач "Активна" для тимчасового відключення без видалення з проекту. Неактивні точки доступу відображаються сірим кольором для візуального розрізнення стану.

3.4 Перегляд результатів моделювання

Режими візуалізації. Система підтримує три основні режими відображення результатів моделювання. Режим "Сила сигналу" показує рівень сигналу у децибелах відносно міліват у кожній точці приміщення. Режим "Зони покриття" відображає області з прийнятним рівнем сигналу для стабільного підключення пристроїв. Режим "Втрати шляху" візуалізує втрати потужності сигналу відносно відстані від джерел.

Налаштування відображення. Натисніть кнопку "Налаштування" у верхній лівій частині екрану для доступу до параметрів візуалізації. Оберіть бажаний режим відображення з доступного списку варіантів. Налаштуйте кольорову схему вибравши стандартну палітру з переходом від синього через зелений до червоного кольору, монохромну схему з градаціями сірого або контрастну схему з різкими переходами між кольорами. Встановіть діапазони

відображення визначивши мінімальні та максимальні значення у децибелах для оптимального контрасту.

Статистична інформація. Нижня частина екрану відображає ключові метрики проекту включаючи відсоток покритої площі приміщення сигналом достатньої потужності, середній рівень сигналу по всій території, кількість активних точок доступу та інформацію про виявлені зони слабого покриття що потребують додаткової уваги.

3.5 Експорт результатів

Збереження зображення. Натисніть кнопку "Експорт" у правій верхній частині екрану та оберіть опцію "Зображення PNG" для створення растрового файлу. Налаштуйте роздільність зображення вибравши стандартні варіанти 1080p або 4K або встановіть власні параметри відповідно до потреб. Натисніть "Зберегти" та оберіть спосіб поширення результату через доступні системні сервіси.

Створення PDF звіту. Оберіть опцію "PDF звіт" у меню експорту для генерації професійного документу. Введіть додаткову інформацію включаючи назву об'єкту, дату проведення планування та коментарі щодо особливостей проекту. Натисніть "Створити звіт" для формування документу з векторною графікою та детальними характеристиками.

Експорт табличних даних. Оберіть опцію "Табличні дані" для створення файлу у форматі CSV. Визначте формат експортованих даних включаючи значення по точках розрахункової сітки, статистику по зонах покриття або параметри точок доступу. Підтвердіть експорт для створення структурованого файлу придатного для подальшого аналізу.

4. Поради та рекомендації

4.1 Оптимальне використання

Точність моделювання. Для досягнення максимальної точності результатів використовуйте коректні типи будівельних матеріалів стін відповідно до реальних конструкцій приміщення. Враховуйте товщину стін при виборі матеріалу оскільки цей параметр впливає на коефіцієнт атенуації. Розміщуйте

точки доступу у реалістичних позиціях з урахуванням можливостей прокладання кабелів та доступу до електроживлення. Регулярно перевіряйте налаштування потужності передачі та частотного діапазону відповідно до специфікацій обладнання.

Продуктивність системи. Для роботи з великими приміщеннями рекомендується використовувати менш деталізовану розрахункову сітку для зменшення обчислювального навантаження. Закрийте інші активні додатки для звільнення оперативної пам'яті та покращення швидкодії. Регулярно зберігайте проекти для запобігання втраті даних при несподіваних збоях системи.

4.2 Типові помилки та методи усунення

Повільний розрахунок теплових карт. У разі тривалого очікування результатів моделювання зменшіть кількість точок доступу до 10-15 одиниць для зниження обчислювальної складності. Спростіть геометрію приміщення видаливши несуттєві архітектурні елементи. Закрийте фонові додатки що споживають системні ресурси.

Неточні результати моделювання. При отриманні нереалістичних результатів перевірте правильність вибору матеріалів стін та їх відповідність реальним конструкціям. Переконайтеся у коректності налаштувань потужності передачі точок доступу згідно з технічними характеристиками обладнання. Використовуйте реалістичні розміри приміщення та масштаб карти.

Проблеми з експортом. При виникненні помилок експорту переконайтеся у наявності достатнього вільного простору на пристрої мінімум 50 МБ для тимчасових файлів. Спробуйте експорт у меншій роздільності для зменшення розміру результуючих файлів. У критичних випадках перезапустіть додаток для очищення системного кешу.

5. Часті запитання

Чи можна використовувати додаток без підключення до інтернету?

Програмне забезпечення повністю функціонує в автономному режимі без потреби в постійному підключенні до мережі після завершення процесу встановлення.

Скільки проектів можна зберегти на пристрої? Кількість проектів обмежується виключно доступним вільним простором на накопичувачі пристрою. Середній розмір одного проекту становить від 1 до 5 мегабайт залежно від складності геометрії.

Чи підтримуються інші стандарти бездротових мереж? На поточний момент додаток орієнтований на стандартні частотні діапазони 2.4 та 5 ГГц відповідно до специфікацій IEEE 802.11.

ОСНОВНІ ПРОГРАМНІ МОДУЛІ СИСТЕМИ

```

swift

// MARK: - Model Layer (Рівень даних)

// Структура для представлення WiFi джерела
struct WiFiSource: Identifiable, Codable {
    let id = UUID()
    var position: CGPoint
    var power: Double // Потужність передавача (10-30 dBm)
    var frequency: WiFiFrequency // 2.4 або 5 ГГц
    var isActive: Bool = true

    enum WiFiFrequency: String, CaseIterable, Codable {
        case freq2_4GHz = "2.4 GHz"
        case freq5GHz = "5 GHz"

        var pathLossExponent: Double {
            switch self {
                case .freq2_4GHz: return 2.0
                case .freq5GHz: return 2.4
            }
        }
    }
}

// Структура для представлення стіни
struct Wall: Identifiable, Codable {
    let id = UUID()
    var startPoint: CGPoint
    var endPoint: CGPoint
    var material: WallMaterial

    enum WallMaterial: String, CaseIterable, Codable {
        case drywall = "Гіпсокартон"
        case brick = "Цегла"
        case concrete = "Бетон"
        case glass = "Скло"
        case wood = "Дерево"

        var attenuationFactor: Double {
            switch self {
                case .drywall: return 3.0
                case .brick: return 8.0
                case .concrete: return 15.0
                case .glass: return 2.0
                case .wood: return 4.0
            }
        }
    }
}

```

Рисунок В.1 - Структура для представлення WiFi джерела та стіни

```

import SwiftUI
import Combine

// MARK: - ViewModel Layer
class SimulatorViewModel: ObservableObject {
    @Published var currentProject: MapProject?
    @Published var savedProjects: [MapProject] = []
    @Published var isCalculating = false
    @Published var heatmapData: [[Double]] = []
    @Published var visualizationMode: VisualizationMode = .signalStrength

    private let storageManager = MapStorageManager()
    private let propagationEngine = SignalPropagationEngine()
    private let exportManager = ExportManager()

    enum VisualizationMode: String, CaseIterable {
        case signalStrength = "Signal Strength"
        case coverageZones = "Coverage Zones"
        case pathLoss = "Path Loss"
        case interference = "Interference"
        case customRange = "Custom Range"
    }

    // Створення нового проекту
    func createNewProject(name: String, scale: Double) {
        let newProject = MapProject(name: name, scale: scale)
        currentProject = newProject
        saveCurrentProject()
    }

    // Додавання WiFi джерела
    func addWiFiSource(at position: CGPoint,
                      power: Double,
                      frequency: WiFiSource.WiFiFrequency) {
        guard var project = currentProject else { return }

        let newSource = WiFiSource(position: position,
                                    power: power,
                                    frequency: frequency)
        project.wifiSources.append(newSource)
        currentProject = project
        calculateCoverage()
    }

    // Основний метод розрахунку покриття
    func calculateCoverage() {
        guard let project = currentProject else { return }
        isCalculating = true

        DispatchQueue.global(qos: .userInitiated).async { [weak self] in
            let coverage = self?.propagationEngine.calculateHeatmap(
                sources: project.wifiSources,
                walls: project.walls,
                gridSize: CGSize(width: 100, height: 100),
                scale: project.scale
            ) ?? []

            DispatchQueue.main.async {
                self?.heatmapData = coverage
                self?.isCalculating = false
            }
        }
    }
}

```

Рисунок В.2 - Структура ViewModel, шар бізнес логіки

```

class SignalPropagationEngine {

    func calculateHeatmap(sources: [WiFiSource],
                        walls: [Wall]) -> [[Double]] {
        let gridSize = 100
        var heatmap: [[Double]] = Array(
            repeating: Array(repeating: -100.0, count: gridSize),
            count: gridSize
        )

        for row in 0..

```

Рисонок В.3 - Матиматична модель

```

import SwiftUI

struct HeatmapView: View {
    let heatmapData: [[Double]]

    var body: some View {
        Canvas { context, size in
            drawHeatmap(context: context, size: size)
        }
    }

    private func drawHeatmap(context: GraphicsContext, size: CGSize) {
        let rows = heatmapData.count
        let cols = heatmapData[0].count

        let cellWidth = size.width / CGFloat(cols)
        let cellHeight = size.height / CGFloat(rows)

        for row in 0..

```

Рисонок В.4 - Візуалізація теплової карти

```
class ExportManager {  
  
    func exportPNG(heatmapData: [[Double]],  
                  projectName: String) -> Data? {  
        let renderer = ImageRenderer(content:  
            VStack {  
                Text(projectName).font(.title)  
                HeatmapView(heatmapData: heatmapData)  
                    .frame(width: 400, height: 400)  
            }  
        )  
        return renderer.pngData  
    }  
  
    func exportCSV(heatmapData: [[Double]]) -> String {  
        var csv = "X,Y,Signal_dBm\n"  
  
        for (row, rowData) in heatmapData.enumerated() {  
            for (col, value) in rowData.enumerated() {  
                csv += "\(col),\(row),\(value)\n"  
            }  
        }  
  
        return csv  
    }  
}
```

Рисонок В.5 - Модуль експорту результатів

АНОТАЦІЯ

Ткачук Б.Р. – Розробка iOS-додатку для моделювання поширення WiFi-сигналу. Рукопис. Кваліфікаційна робота на здобуття освітнього ступеня «магістр» за спеціальністю 122 Комп'ютерні науки, освітня програма «Комп'ютерні науки та інформаційні технології». – Волинський національний університет імені Лесі Українки. – 2025.

Дослідження присвячено розробці доступного мобільного інструментарію для математичного моделювання процесів поширення WiFi-сигналів у закритих архітектурних просторах. Проведено комплексний аналіз існуючих програмних рішень у сфері планування бездротових мереж та виявлено істотний розрив між високовартісними професійними системами та спрощеними споживчими інструментами. Досліджено теоретичні основи поширення електромагнітних хвиль у приміщеннях та обґрунтовано застосування математичної моделі Log-Distance Path Loss для мобільної реалізації. Спроековано та реалізовано повнофункціональний iOS-застосунок з використанням Swift, SwiftUI та архітектурного патерну MVVM для інтерактивного планування покриття бездротових мереж. Система забезпечує створення планів приміщень, розміщення точок доступу, розрахунок поширення радіосигналу в режимі реального часу та візуалізацію результатів через теплові карти. Валідація математичної моделі показала середню абсолютну похибку 6,8 дБ, а тестування з користувачами підтвердило високу зручність використання з оцінкою 8,5 балів з 10.

Ключові слова: iOS-розробка, Swift, WiFi-моделювання, поширення радіосигналу, мобільні додатки, SwiftUI, MVVM, теплові карти.

ABSTRACT

Tkachuk B.R. – Development of an iOS Application for Modeling Wi-Fi Signal Propagation. Manuscript. Qualification work for the degree of "Master" in the field of Computer Science, educational program "Computer Science and Information Technologies." – Lesya Ukrainka Volyn National University. – 2025.

The study addresses the development of accessible mobile toolset for mathematical modeling of WiFi signal propagation in enclosed architectural environments. A comprehensive analysis of existing software solutions revealed a significant gap between expensive professional systems and simplified consumer tools with limited capabilities. Theoretical foundations of electromagnetic wave propagation were investigated and the Log-Distance Path Loss mathematical model was justified for mobile implementation. A fully functional iOS application was designed and implemented using Swift, SwiftUI, and MVVM architectural pattern for interactive wireless network coverage planning. The system provides functionality for creating floor plans, placing access points, calculating signal propagation in real-time, and visualizing results through heat maps. Mathematical model validation demonstrated a mean absolute error of 6.8 dB, while user testing confirmed high usability with an average score of 8.5 out of 10.

Keywords: iOS development, Swift, WiFi modeling, radio signal propagation, mobile applications, SwiftUI, MVVM, heat maps.