

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА**

**Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»**

Кафедра інформаційних систем в економіці

**ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА
«Системи штучного інтелекту»**

галузь знань 12 Інформаційні технології
спеціальність 122 Комп'ютерні науки

Форма навчання: очна (денна)

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему: **«ІНТЕЛЕКТУАЛЬНА СИСТЕМА ЗБОРУ ТА СИСТЕМАТИЗАЦІЇ
ДАНИХ З МЕРЕЖІ ПРИСТРОЇВ МОНІТОРИНГУ.»**

здобувача _____ Д. С. Білокопитов _____
(ПІБ, підпис)

Науковий керівник: _____ д.т.н., професор А. І. Шевченко _____
(науковий ступінь, учене звання, ПІБ)

(підпис)

**Робота допущена до захисту перед
екзаменаційною комісією з атестації здобувачів
вищої освіти (ЕК)**

Завідувач кафедри: _____ Б. О. Тішков _____
(науковий ступінь, учене звання, ПІБ)

(підпис)

Київ 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КІЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА

«Системи штучного інтелекту»

галузь знань 12 Інформаційні технології
спеціальність 122 Комп'ютерні науки

ПОГОДЖЕНО:
Керівник проектної групи (гарант)
освітньо-професійної програми

Рамазанов С.К.
«___» _____ 2024р.

ЗАТВЕРДЖУЮ:
Завідувач кафедри інформаційних
систем в економіці

Тішков Б.О.
«___» _____ 2024р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувача вищої освіти _____ Д. С. Білокопитов
(прізвище, ім'я, по батькові)

_____ очної (денної) _____ форми навчання
очної (денної), заочної, дистанційної

на підготовку кваліфікаційної магістерської роботи
на тему: «ІНТЕЛЕКТУАЛЬНА СИСТЕМА ЗБОРУ ТА СИСТЕМАТИЗАЦІЇ ДАНИХ З МЕРЕЖІ
ПРИСТРОЇВ МОНІТОРИНГУ»

Тему затверджено наказом ректора Університету від «__» _____ 2024 р. № _____

Кваліфікаційна магістерська робота виконується на матеріалах

План кваліфікаційної магістерської роботи

Розділ I ТЕОРЕТИЧНИЙ РОЗДІЛ. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО
СТВОРЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ЕКСПЕРТНОЇ СИСТЕМИ.

(назва розділу)

Розділ II АНАЛІТИЧНИЙ РОЗДІЛ. ХАРАКТЕРИСТИКА ЕКСПЕРТНОЇ СИСТЕМИ ТА
ПОСТАНОВКА ЗАДАЧІ.

(назва розділу)

Розділ III КОНСТРУКТИВНИЙ РОЗДІЛ. РОЗРОБЛЕННЯ ПРОЕКТНИХ РІШЕНЬ.
(назва розділу)

Об'єкт дослідження: Система моніторингу GSM сигналу.

Предмет дослідження: Процес проектування системи моніторингу та подальшої обробки GSM сигналу.

Мета кваліфікаційної магістерської роботи: Проектування системи прийому та подальшої обробки GSM сигналу, яка допоможе оптимізувати прийняття рішень у військовій справі. Та значно знизити кількість загиблих солдат. Для досягнення цієї мети будуть використані сучасні методи та інструменти програмування, методи аналізу даних, а також апаратні можливості сучасних датчиків та мікроконтролерів.

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:
У розділі I Розглядається дослідження предметної області на основі зібраної інформації, аналіз існуючих СШ й інтелектуальних систем предметної області, постановка проблеми та постановка задачі, обґрунтування вибору підходів і технологій для проектування та створення експертної системи та її компонентів.

У розділі II Розглядається характеристика об'єкта дослідження та постановка задачі.

У розділі III Розглядається моделювання та проектування 3D моделі для експертної системи, моделювання та проектування бази даних для системи прийняття рішень, розроблення користувацького інтерфейсу, елементи та структура експертної системи, моделювання та проектування апаратної реалізації для експертної системи та обґрунтування вибору розподіленої архітектури.

Завдання підготував науковий керівник _____
(підпис) (ініціали, прізвище)

« ____ » _____ 202_р.

Завдання одержав здобувач _____
(підпис) (ініціали, прізвище)

« ____ » _____ 202_р.

Реферат

Кваліфікаційна магістерська робота містить 85 сторінок, 4 таблиці, 28 рисунків, список використаних джерел з 20 найменувань, додатки.

«Інтелектуальна Система збору та систематизації даних з мережі пристроїв моніторингу»

Об'єктом дослідження кваліфікаційної магістерської роботи виступає система моніторингу GSM сигналу.

Предметом дослідження є процес проектування системи моніторингу та подальшої обробки GSM сигналу.

Мета і завдання дослідження. Основною метою кваліфікаційної магістерської роботи є проектування системи прийому та подальшої обробки GSM сигналу, яка допоможе оптимізувати прийняття рішень у військовій справі. Та значно знизити кількість загинутих солдат. Для досягнення цієї мети використовуються сучасні методи та інструменти програмування, методи аналізу даних, а також апаратні можливості сучасних датчиків та мікроконтролерів.

Відповідно до поставленої мети визначені такі *завдання*:

- дослідити предметну область, виконати аналіз існуючих систем з обробки GSM сигналів.
- представити обґрунтування вибору підходів і технологій для створення системи моніторингу;
- визначити структуру та представити характеристику системи GSM моніторингу;
- описати методи та моделі в системі інтелектуальної взаємодії;
- спроектувати базу даних для системи моніторингу GSM сигналів;
- спроектувати модулі з шифрування даних для передачі та зберігання інформації для системи збору та систематизації даних з мережі пристроїв моніторингу;
- описати програмне забезпечення для системи інтелектуальної взаємодії;
- описати технічне забезпечення для системи інтелектуальної взаємодії;
- описати методи для обміну пакетами даних між апаратною та

програмною складовою системи;

- розробити 3D модель яка забезпечує можливість використання інтелектуальної системи збору та систематизації даних з мережі пристроїв моніторингу;

- описати та розробити апаратну реалізацію яка забезпечує роботу інтелектуальної системи зі збору та систематизації даних з мережі пристроїв моніторингу;

- представити результати реалізації системи інтелектуальної взаємодії зі збору та систематизації даних з мережі пристроїв моніторингу.

Теоретична, методична та практична значущість отриманих результатів. Під час дослідження було досліджено предметну область, проаналізовано варіанти зв'язку між апаратною та програмною складовою системи моніторингу, розглянуто варіанти корекції положення антени у просторі, варіанти доступу до бази даних системи, визначено перспективність удосконалення процесу виявлення сигналів що випромінюються.

Практичні результати дослідження полягають у вдосконаленні систем зі збору та систематизації даних з мережі пристроїв моніторингу, що призведе до підвищення ефективності у рамках військової галузі.

Рік виконання кваліфікаційної магістерської роботи – 2024.

Рік захисту роботи – 2024.

Ключові слова: Моніторинг, антена, інформація, інтерфейс, мікроконтролер, шифрування, інформаційна система.

ВІДГУК

про кваліфікаційну магістерську роботу
здобувача навчально-наукового інституту «Інститут інформаційних технологій
в економіці»
освітньо-професійної програми «Системи штучного інтелекту»
Білокопитова Дмитра Сергійовича
на тему «Інтелектуальна система збору та систематизації даних з мережі
пристроїв моніторингу»

1. Актуальність теми: актуальність є обґрунтовано високою в умовах сучасних викликів державної безпеки та захисту від вторгнення країни-агресора. Застосування широкого спектру систем радіоелектронної боротьби вимагає нових підходів та розробок щодо оперативної аналітики та протидії ворожим засобам. Створення мобільних систем збору та передачі даних із їх подальшою обробкою за допомогою методів штучного інтелекту є перспективним напрямом для вирішення таких завдань.

2. Позитивні риси кваліфікаційної магістерської роботи: в рамках роботи виконано комплекс інженерно-технічних та дослідницьких робіт: аналіз апаратних продуктів моніторингу радіо середовища, проектування рухомої платформи та моделювання її елементів, створення реляційної моделі даних СУБД, інтерфейсу користувача системи, налаштування серверу обробки даних. Виконані завдання дозволили створити апаратно-програмний прототип інтелектуальної системи збору даних з мережі пристроїв моніторингу.

3. Наявність самостійних розробок автора: програмна реалізація сервісу прогнозування руху джерела радіо сигналу на основі моделі штучної нейронної мережі, інтерфейс відображення стану радіо середовища, апаратна схема та програмні модулі керування поворотом антени станції моніторингу.

4. Цінність теоретичних висновків та практичних рекомендацій: аналіз антенних систем та апаратних засобів збору даних дозволив знайти рішення для аналізу спектру частот від 35 Гц до 6 ГГц. Розглянуті методи триангуляції та позиціонування джерела випромінювання реалізовано програмно із використанням прикладного програмного інтерфейсу розробників апаратних засобів збору даних.

5. Наявність недоліків: у роботі реалізовано лише один із низки методів штучного інтелекту для обробки даних, дослідження прогнозування руху джерела радіо сигналу альтернативними методами може підвищити ефективність та практичну цінність розробки.

6. Загальна оцінка кваліфікаційної магістерської роботи та її допущення до захисту перед ЕК: завдання роботи виконано на високому рівні у визначені терміни, що засвідчує відмінний рівень підготовки здобувача. Рекомендується до захисту перед ЕК.

Науковий керівник професор, д. т. н., чл.-кор. НАН України



Шевченко А. І.

РЕЦЕНЗІЯ

на кваліфікаційну магістерську роботу здобувача вищої освіти

Білокопитова Дмитра Сергійовича

Тема «Інтелектуальна система збору та систематизації даних з мережі пристроїв моніторингу»

Актуальність теми кваліфікаційної магістерської роботи і доцільність її розроблення: Враховуючи характер бойових дій із застосуванням низки засобів радіо передачі даних та боротьби, актуальність роботи для держави не викликає сумнівів. Розроблений програмний додаток із інтерфейсом користувача є доцільним для використання на сучасних мобільних платформах моніторингу.

Якість проведеного дослідження:

Проведено детальний аналіз актуального стану вирішення проблеми моніторингу радіо середовища портативними розподіленими засобами збору даних. Для програмної реалізації обрано сучасні засоби розробки, апаратна платформа базується на поширеній компонентній базі.

Позитивні риси кваліфікаційної магістерської роботи:

Результатом роботи є апаратно-програмний фреймворк інтелектуальної системи збору даних та апаратно-програмний прототип пристрою моніторингу, що може стати платформою для подальших досліджень із практичного застосування СШІ з метою виявлення та класифікації об'єктів у радіо середовищі.

Зауваження: у роботі використано дві станції моніторингу для позиціонування джерела радіо сигналу, для триангуляції включно із напрямом пропонується використання трьох станцій. Відсутній аналіз аналогічних засобів радіо моніторингу військового призначення.

Практичне значення висновків і рекомендацій:

У роботі продемонстровано як знання і навички здобувача, так і результати аналізу наукових джерел, отримані під час виконання роботи. Обрана тема дозволила зрозуміти принципи методів моніторингу GSM сигналів та реалізувати власний підхід до розробки портативної системи збору даних.

Місце роботи та посада рецензента
науковий ступінь учене звання
Старший науковий співробітник
Інституту проблем штучного інтелекту
МОН України і НАН України
д.т.н., проф.



Слюсар В.І.

підпис засвідчую _____

ЗМІСТ

ВСТУП	10
1 ТЕОРЕТИЧНИЙ РОЗДІЛ. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО СТВОРЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ЕКСПЕРТНОЇ СИСТЕМИ	13
1.1 Дослідження предметної області. Збір інформації та вивчення матеріалів з теми кваліфікаційної магістерської роботи	13
1.2 Аналіз існуючих СШІ й інтелектуальних систем предметної області.....	26
1.3 Постановка проблеми та формування задач	26
1.4 Обґрунтування вибору підходів і технологій для проектування та створення експертної системи та її компонентів.....	28
2 АНАЛІТИЧНИЙ РОЗДІЛ. ХАРАКТЕРИСТИКА ЕКСПЕРТНОЇ СИСТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ.....	30
2.1 Характеристика об'єкта дослідження	30
2.2 Постановка задачі	31
3 КОНСТРУКТИВНИЙ РОЗДІЛ. РОЗРОБЛЕННЯ ПРОЕКТНИХ РІШЕНЬ 33	
3.1 Моделювання та проектування 3D моделі для експертної системи.....	33
3.2 Моделювання та проектування бази даних для системи прийняття рішень	36
3.3 Розроблення користувацького інтерфейсу. Елементи та структура	41
3.4 Моделювання та проектування апаратної реалізації для експертної системи.....	43
3.5 ПРОЕКТУВАННЯ ЗАБЕЗПЕЧУВАЛЬНИХ ПІДСИСТЕМ ЕКСПЕРТНОЇ СИСТЕМИ	47
3.5.1 Розгортання експертної системи та налаштування параметрів	50
3.5.2 Обґрунтування вибору розподіленої архітектури	54
ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	62
ДОДАТКИ.....	64
Додаток А Фрагменти програмної реалізації інтервейсу користувача.....	64
Додаток Б Фрагменти програмного коду стилів.....	66
Додаток В Фрагменти програмного коду форми авторизації.....	71
Додаток Г Фрагменти програмного коду радару.....	72
Додаток Д Фрагменти програмного коду управління антеною GSM	74
Додаток Е Фрагменти програмного коду для системи ШІ	83

ВСТУП

Система збору та систематизації даних з мережі пристроїв моніторингу – як багато в цьому реченні, для когось це щось невідоме, але тільки не для випускників з галузі інформаційних систем. Розвиваючись у відповідній галузі, минулі випускники вищу, а згодом і висококваліфіковані фахівці, якими стають подолавши тернистий шлях, здатні подарувати світу вражаючі відкриття та ідеї, які здатні вплинути на велику кількість галузей, такі як робототехніка, космічна галузь, військова справа та інші. Варто пам'ятати, що розвиток таких галузей не обійде боком і економічна складова країни яка активно розвиває ці ключові сегменти. Розвиваючи напрям з розробки систем збору та систематизації даних з мережевих пристроїв моніторингу, дає простір для розвитку супутніх питань які можна порушити, а саме: етичність використання та створення штучного інтелекту який здатен взаємодіяти з зовнішніми датчиками моніторингу навколишнього середовища, залежність всього людства від технологій, всюдисущі корпорації, які по суті стають вищими за уряд в ієрархії держави, соціальну нерівність між людьми, віртуальна реальність, що замінює багатьом те, що оточує нас, список можна продовжувати нескінченно, але основна суть говорить про те, що всі ці речі з кожним днем для нас стають все реальнішими і реальнішими, і при цьому так само залишаються чимось з розряду фантастики.

Комплексна система спостережень, збору, обробки, систематизації та аналізу інформації про стан навколишнього середовища, яка дає оцінку і прогнозує його зміни, розробляє обґрунтовані рекомендації для прийняття управлінських рішень, що являє собою не чим іншим як системою збору та систематизації даних з мережі пристроїв моніторингу. Такі системи здатні значно та позитивно вплинути на країну у різних аспектах.

У військовій справі система збору та систематизації даних з мережі пристроїв моніторингу відіграє одну з важливих ролей, та розкриває всі свої переваги у час коли в країні введено військове положення. Основним цінним ресурсом вважається людське життя, будь то військового який захищає державу

або цивільного який може виконувати гуманітарні місії на фронті. Таким чином можна сказати що система збору та систематизації даних з мережі пристроїв моніторингу може зберігти купу людських життів шляхом відстеження пересування ворожих сил і теоретичний розрахунок їхнього місцезнаходження у майбутньому. Використовуючи такі системи моніторингу військові будуть мати не тільки можливість опрацювати свою логістику і шляхи пересування, але і налаштувати лінії фронту та отаки на шляху проходження ворогів.

Система збору та систематизації даних з мережі пристроїв моніторингу, може використовувати різноманітні реалізації та мають змогу приймати необхідні потоки даних, в залежності від датчиків – прийомників інформації. Одним з таких видом вхідних даних виступає GSM (Global System for Mobile Communications) сигнал, який моніториться відповідним обладнанням яке складається з апаратної частини та спеціального програмного забезпечення. GSM моніторинг у військовій галузі використовується для різноманітних цілей, включаючи розвідку, нагляд, безпеку та комунікації. Однак, важливо відзначити, що такі операції часто мають делікатний характер і регулюються національними законами та міжнародними угодами. Варто зазначити що військові можуть використовувати GSM моніторинг тільки і тільки для стеження за рухом або комунікацією потенційних противників. Це може включати відстеження викликів, текстових повідомлень та інших форм комунікації зі сторони ворожих сил. З іншого боку GSM моніторинг може бути використаний для забезпечення безпеки військових об'єктів та персоналу, виявляючи неавторизовану активність у критичних зонах, що дає можливість завчасно зробити додаткові перевірки на наявність потенційних ворожих сил (контррозвідка), та у подальшому прийняти рішення для блокування спроб прослуховування або інших ворожих дій проти військових комунікаційних мереж.

Військові командири можуть використовувати GSM моніторинг для координації та управління полем бою, включаючи рух військ, логістику та

зв'язок, а також підтримувати миротворчі та гуманітарні місії (військові організації часто використовують зашифровані канали зв'язку, щоб захистити свої комунікації від прослуховування). Для підтримки цивільних, GSM моніторинг може бути корисним у миротворчих операціях для відстеження руху цивільного населення або допомоги у координації гуманітарної допомоги.

Використання GSM моніторингу у військовій галузі має супроводжуватися високим рівнем безпеки та конфіденційності, а також відповідати юридичним та етичним нормам. Оскільки це чутлива тема, конкретні деталі та методи часто залишаються недоступними для широкої публіки.

Актуальність теми з відтворення систем збору та систематизації даних з мережі пристроїв моніторингу має значну роль для спеціалістів з військової справи та країни в цілому, через можливість здійснювати більш продумані та ефективні операції з наступу або оборони від ворожих сил з боку країни агресора. Таким чином ефективно проведена військова операція здатна значно зменшити втрати зі стоорони військових.

Об'єктом дослідження виступає система моніторингу GSM сигналу.

Предметом дослідження є процес проектування системи моніторингу та подальшої обробки GSM сигналу.

Мета даного дипломного проекту є проектування системи прийому та подальшої обробки GSM сигналу, яка допоможе оптимізувати прийняття рішень у військовій справі. Та значно знизити кількість загинутих солдат. Для досягнення цієї мети будуть використані сучасні методи та інструменти програмування, методи аналізу даних, а також апаратні можливості сучасних датчиків та мікроконтролерів. У даному дипломному проекті будуть розглянуті такі питання, як аналіз вирішуваних задач, дослідження предметної області,

постановка проблеми та формування задачі, обґрунтування вибору підходів і технологій для проектування системи, вибір методів та засобів розробки системи.

1 ТЕОРЕТИЧНИЙ РОЗДІЛ. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО СТВОРЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ЕКСПЕРТНОЇ СИСТЕМИ

В першому теоретичному розділі дипломного проекту буде проведений аналіз предметної області шляхом збору та систематизації інформаційних ресурсів які дозволяють поставити задачу та оприділитися з варіантами підходів для реалізації експертної системи. Також перший теоритичний розділ описує існуючі рішення у системах штучного інтелекту для експертної області системи.

Постановка проблеми та вибір підходів для реалізації поставленної задачі, проводиться на основі здобутої інформації і ретельної її сестиматизації. Описаний розділ з постановки проблеми та опис підходів для її реалізації допоможе знайти оптимальне рішення для реалізації експертної системи.

Окрім того, у цьому розділі будуть проаналізовані технології та підходи, які дозволяють реалізувати експертну систему та її складові.

У підсумку, перший теоретичний розділ дипломного проекту дозволить добре розкрити види реалізацій систем в сфері GSM моніторингу, та проаналізувати і описати основні підходи та шляхи їх вдосконалення чи модифікації.

1.1 Дослідження предметної області. Збір інформації та вивчення матеріалів з теми кваліфікаційної магістерської роботи

Більшість людей користується мобільними телефонами у своєму житті майже щодня. Відповідно така тенденція впливає на збільшення кількості абонентів з кожним днем, так само як і площі мережевого покриття яке здатне забезпечити надійну роботу для пристроїв мобільного зв'язку. Мобільна передача інформації, являє собою спеціальні стандарти які рухаються в крок з розвитком технологій, що приводить до їх змін на більш новітні, а користувачі мобільного інтернету у свою чергу теж спонукають створювати та розвивати

більш новіші, а головне надійні стандарти мобільного зв'язку шляхом створенням конкурентної боротьби серед постачальників відповідної послуги.

Мобільний зв'язок з'явився досить давно. Ще в 40-х роках ХХ століття почалися дослідження з метою створення мережі рухомого зв'язку. І скрізь довгі роки, а саме у 1992 році, слід за стандартами мобільних мереж NMT-450, AMPS, ETACS, D-AMPS і NMT-900 у Німеччині запускається сотовий зв'язок з урахуванням стандарту GSM (Groupe Special Mobile).

В стандарті GSM аудіосигнали перетворюються з аналогової форми в цифрову перед тим, як вони будуть передані. Це перетворення є ключовим для забезпечення високого рівня безпеки, який є характерним для GSM мереж.

Загальна безпека у системах мобільного зв'язку GSM базується на комбінації технічних, організаційних і юридичних аспектів. Для досягнення максимально ефективного захисту, ці елементи мають працювати у координації один з одним.

В стандарті GSM досягається висока степеь безпеки передачі повідомлень яке здійснюється шляхом шифрування повідомлень по алгоритму шифрування з відкритим з відкритим ключем RSA (Rivest–Shamir–Adleman).

Загалом система зв'язку, що діє у стандарті GSM, розрахована на її використання у різних сферах. Вона надає користувачам широкий діапазон послуг та можливість застосовувати різноманітне обладнання для передачі мовних повідомлень та даних, викликових та аварійних сигналів, підключатися до телефонних мереж загального користування (PSTN), мереж передачі даних (PDN) та цифрових мереж з інтеграцією служб (ISDN). [1]

Стандарт GSM містить два класи служб: основні служби та телеслужби. Основні служби забезпечують: передачу даних (асинхронно) у дуплексному режимі зі швидкостями 300, 600, 1200, 2400, 4800 та 9600 біт/с через телефонні мережі загального користування; передачу даних (синхронно) у дуплексному режимі зі швидкостями 1200, 2400, 4800 та 9600 біт/с через телефонні мережі загального користування, комутовані мережі передачі даних загального

користування (CSPDN) та ISDN; доступ за допомогою адаптера до пакетної асинхронної передачі даних зі стандартними швидкостями 300-9600 біт/с через комутовані мережі пакетної передачі даних загального користування (PSPDN), наприклад, Datex-P; синхронний дуплексний доступ до мережі пакетної передачі даних із стандартними швидкостями 2400-9600 біт/с [2].

У структурі GSM суворо визначено часові характеристики огинаючої сигналу, випромінюваного пакетами на каналному часовому інтервалі TDMA кадру, та спектральна характеристика сигналу. Тимчасова маска обгинальної для сигналів, що випромінюються на інтервалі АВ повного TDMA кадру, показана на рис. 1.1, а маска огинаючої для сигналів NB, FB, DB і SB повного TDMA кадру - на рис. 1.2. Різні форми огинаюч их випромінюваних сигналів відповідають різним тривалостям інтервалу АВ (88 біт) по відношенню до інших вказаним інтервалам повного TDMA кадру (148 біт). Норми на спектральну характеристику випромінюваного сигналу показано на рис. 1.3 [1].

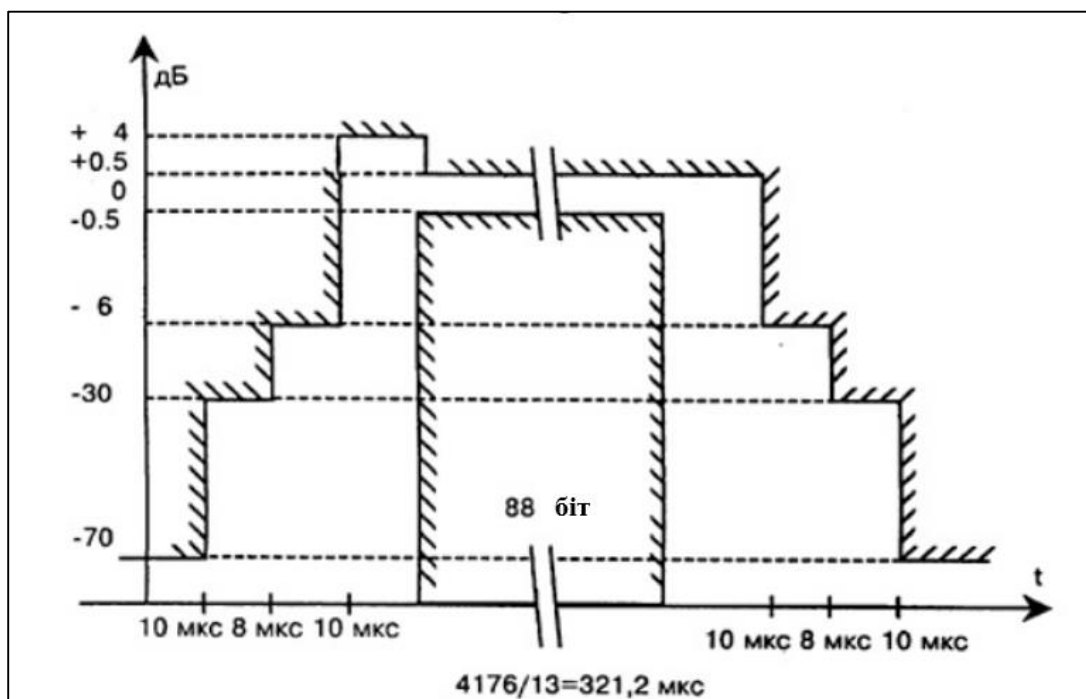


Рисунок 1.1 – Тимчасова маска обгинальної для сигналів

(Джерело: <http://glushilka.narod.ru/Jam/GSM.pdf>)

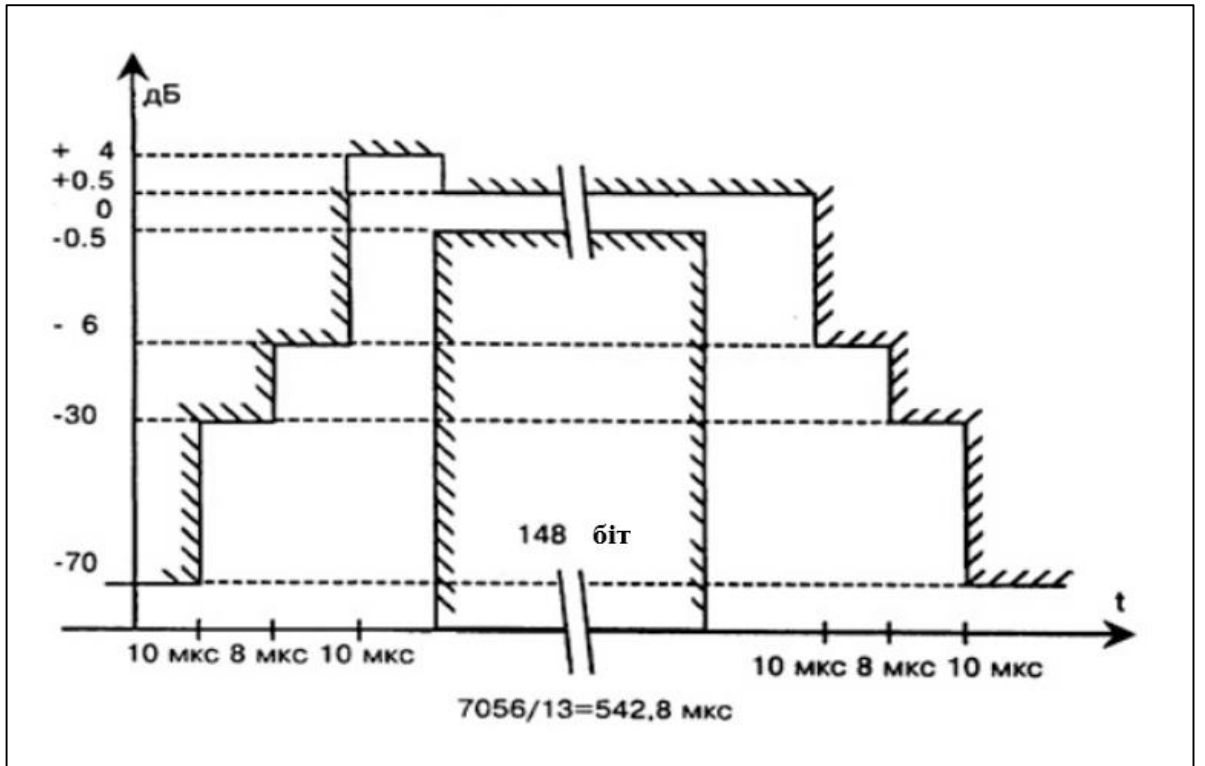


Рисунок 1.2 – Маска огинаючої для сигналів

(Джерело: <http://glushilka.narod.ru/Jam/GSM.pdf>)

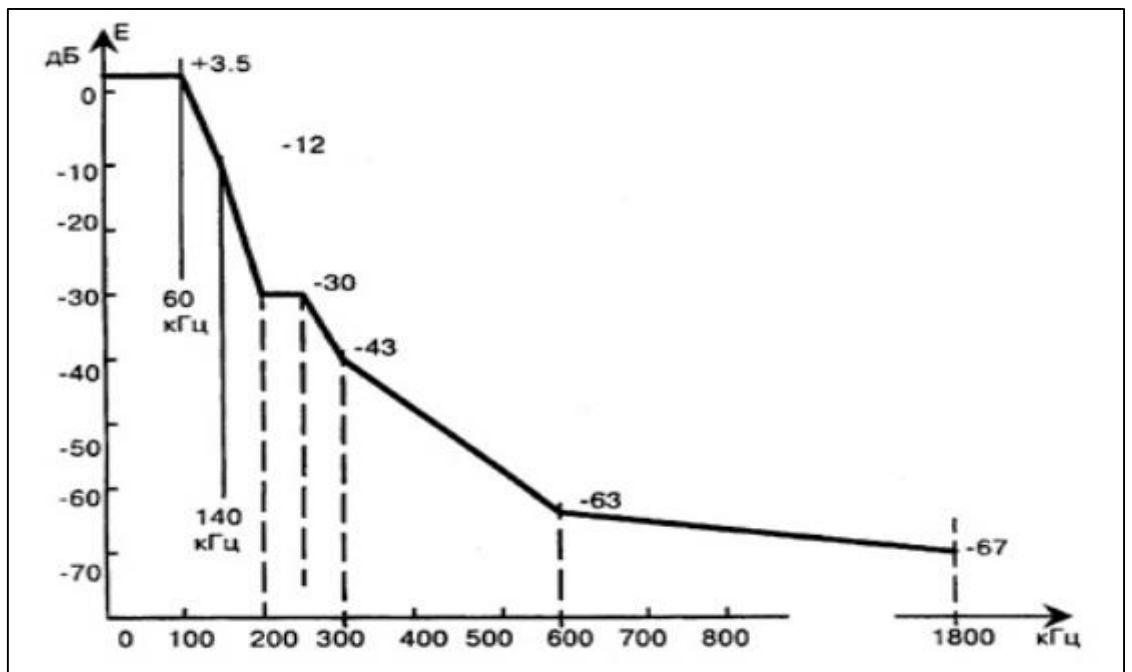


Рисунок 1.3 – Норми на спектральну характеристику сигналу

(Джерело: <http://glushilka.narod.ru/Jam/GSM.pdf>)

Одна з особливостей формування сигналів у стандарті GSM - використання повільних стрибків по частоті у процесі сеансу зв'язку. Головне призначення таких стрибків (SFH - Slow Frequency Hopping) - забезпечення частотного рознесення в радіоканалах, що функціонують в умовах багатопроменевого поширення радіохвиль. SFH використовується у всіх рухомих мережах, що підвищує ефективність кодування та перемежування при повільному русі абонентських станцій. Принцип формування повільних стрибків по частоті полягає в тому, що повідомлення, що передається у виділеному абоненту тимчасовому інтервалі TDMA кадру (577 мкс), у кожному наступному кадрі передається (приймається) на нову фіксовану частоту. Відповідно до структури кадрів час для перебудови частоти складає близько 1 мс.

У процесі стрибків по частоті постійно зберігається дуплексне рознесення 45 МГц між каналами прийому і передачі. Всім активним абонентам, що знаходяться в одній соті, ставляться у відповідність ортогональні формують послідовності, що виключає взаємні перешкоди при прийомі повідомлень абонентами в соті. Параметри послідовності перемикавання частот (частотно-часова матриця та початкова частота) призначаються кожній рухомій станції у процесі встановлення каналу. Ортогональність послідовностей перемикавання частот в соті забезпечується початковим частотним зрушенням однієї і тієї ж (за алгоритмом формування) послідовності. У суміжних стільниках використовуються різні формують послідовності.

В основі безпеки GSM лежать три алгоритми, які є офіційно закритими, тобто. секретними:

- A3 - алгоритм аутентифікації;
- A8 - алгоритм генерації ключа шифрування для сеансу зв'язку;
- A5 – алгоритм шифрування сигналу у процесі сеансу зв'язку.

Аутентифікація – уточнення автентичності абонента. Механізм автентифікації полягає в тому, що мережа GSM надсилає запит із випадковим.

числом на мобільну станцію (телефон абонента), мобільна станція шифрує це число алгоритмом АЗ, використовуючи ключ шифрування, який у свою чергу записано на SIM-карті абонента (модуль автентичності абонента). Зашифрований запит відправляється від мобільної станції назад оператору. Отримана відповідь перевіряється на правильність: оператор проводить таке саме шифрування випадкового числа, відправленого на мобільну станцію. Якщо отриманий результат збігається з відповіддю з мобільної станції, вважається, що автентифікація пройшла успішно. Відповідь, отримана після роботи алгоритму АЗ, потрібна для визначення ключа шифрування. Цей ключ дозволяє передавати все повідомлення у режимі захисту інформації. Зашифрована відповідь надходить алгоритм А8, який у свою чергу за допомогою деяких перетворень обчислює ключ. Подібна дія провадиться і оператором, що забезпечує ідентичність ключа на стороні оператора та стороні мобільної станції [3].

Поряд із випадково визначеним числом, мережа також надсилає на мобільний пристрій спеціальну числову послідовність, яка використовується як ключ шифрування. Ця послідовність тісно корелює з актуальним значенням ключа, що допомагає запобігти створенню помилкового ключа. Мобільний пристрій зберігає це число та інкорпорує його у перше повідомлення, яке відправляється назад у мережу, гарантуючи таким чином безпеку комунікації.

Для того щоб отримати інформацію про закриті дані абонента, необхідно мати досить складно обладнання, яке б дозволяло отримати доступ до обладнання абонента таким чином, щоб абонент сам про це не здогадувався. Таке обладнання є дорогим та доступно аж ніяк не кожній людині.

Також, хотілося б помітити, що доступ до закритих даних є за законом неправомірним діянням і розглядається як шахрайська дія. Проводити подібні операції з отримання особистих даних абонента можна лише у випадках, пов'язаних із силовими органами країни. Тобто. на підставі постанови прокурора або суду, повністю законної, оператор має право надати особисту інформацію абонента третім особам у випадках, коли це є необхідним [4].

Покриття певної місцевості мережовим зв'язком забезпечується за рахунок розподілу приймальних пристроїв по її площі. Найчастіше вони являють собою кілька спрямованих антен білого кольору, а також невелику будівлю, куди тягнуться дроти (рис. 1.4.). Так ось, у термінології GSM такі комплекси називаються базовими станціями (BTS) і можуть складатися з кількох приймальних пристроїв - трансіверів (TRX - Transmitter/Receiver).



Рисунок 1.4 – приймальні пристрої (BTS)

(Джерело: <https://static.promediatekhnologi.id/crop/0x0:0x0/750x500/webp/photo/2022/03/11/1223523699.jpg>)

Мережвий зв'язок працює за принципом поділу загальної зони покриття на невеликі ділянки, які називаються стільниками. Кожна стільникова станція (БС) покриває один або кілька таких сотових комірок. БС має кілька приймачів у кожному секторі, кожен з яких випромінює сигнал на своїй частоті (рис. 1.5.).

Стільники можна класифікувати за масштабом території, що покривається: макросоти (до 35 км), звичайні стільники (до 5 км), мікросоти (до 1 км), пікосоти (до 300 метрів) і фемтосоти (до кількох десятків метрів).

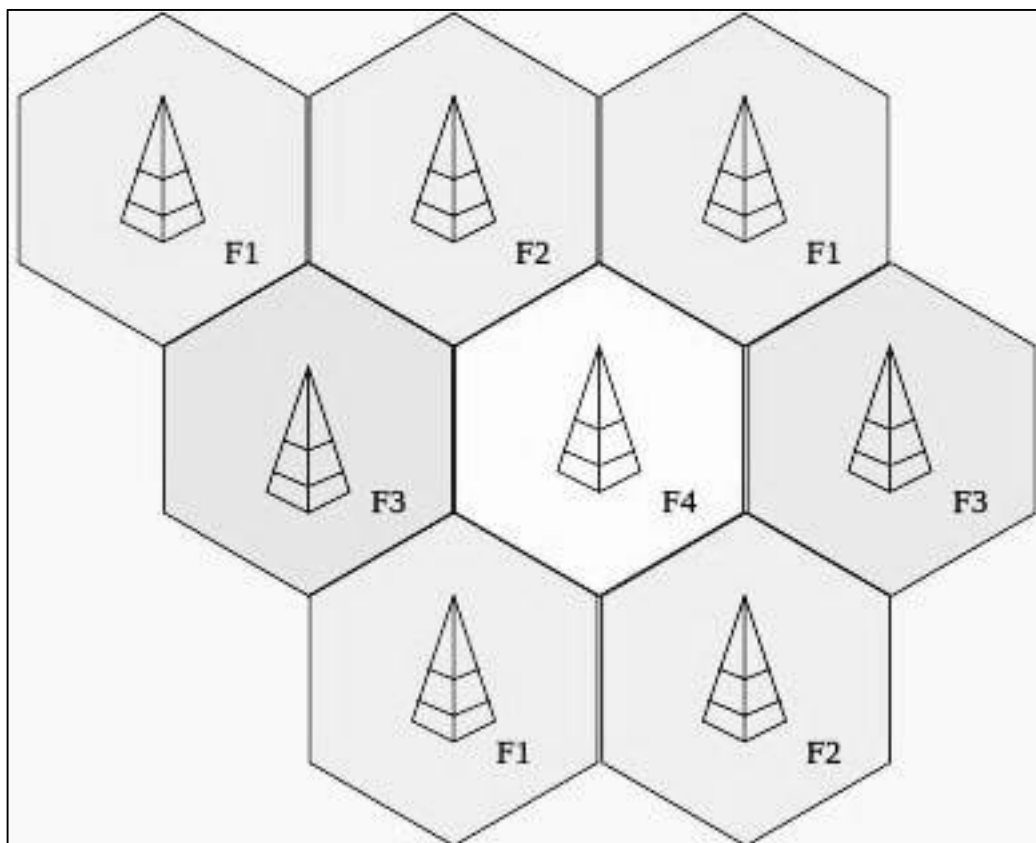


Рисунок 1.5 – стільникові осередки

(Джерело: https://uk.wikipedia.org/wiki/Стільниковий_зв'язок#/media/Файл:Frequency_reuse.svg)

Використовуючі вище перераховані теоретичні відомості, можна плавно перейти до методів які дозволяють отримувати інформацію яка розповсюджується в середині стільникових осередків у вигляді GSM сигналу, який проявляє свою активність при обміні даними серед користувачів, будь то користування інтернетом, або мобільний зв'язок. Таким чином спеціальні підрозділи (наприклад служби безпеки тієї чи іншої країни) можуть через судові рішення отримати дозвіл на отримання кодів доступу та розшифровки GSM сигналу у ефірі моніторингу, які надають представники надавання

мобільного зв'язку. Але підстави для надання подібного доступу до приватної інформації користувачів мережі GSM повинні бути чітко аргументовані зі сторони яка робить запит на моніторинг. До деяких з найбільш поширених підстав для надання такого доступу можна віднести наступне:

- Судовий Наказ: Правоохоронні органи можуть отримати судовий наказ, який вимагає від оператора мобільного зв'язку надати доступ до певної інформації користувача для розслідування злочинів або в інших правових цілях.
- Запити від Правоохоронних Органів: У випадку розслідування злочинів або підозри в терористичній діяльності, правоохоронні органи можуть запросити інформацію про користувача мережі GSM.
- Національна Безпека: У випадках, коли існує загроза національній безпеці, уряди можуть мати право отримувати доступ до телефонних записів, даних про місцезнаходження та іншої інформації.
- Запити від Спеціальних Служб: Спецслужби можуть мати легальний доступ до даних користувачів у рамках своїх повноважень, якщо це пов'язано з питаннями безпеки держави або боротьби з тероризмом.
- Згода Користувача: У певних випадках користувачі можуть надати згоду на обробку своїх персональних даних, наприклад, для отримання індивідуалізованих послуг або участі в маркетингових дослідженнях.
- Законодавство про Захист Даних: Закони про захист даних, такі як Загальний регламент захисту даних (GDPR) у Європейському Союзі, визначають чіткі умови для обробки особистих даних та надання доступу до них.

У всіх випадках, надання доступу до приватної інформації користувачів повинно бути обмежене мінімально необхідним обсягом та виконуватися з дотриманням відповідних законів та процедур. Порушення цих умов може призвести до юридичної відповідальності для осіб або організацій, які отримують або використовують дані. Також варто пам'ятати що порушення умов

і правил, що регламентують діяльність у сфері телекомунікацій та користування радіочастотним ресурсом України, передбачену ліцензіями, дозволами, повідомленням про початок здійснення діяльності у сфері телекомунікацій, тягне за собою накладення штрафу на посадових осіб підприємств і організацій усіх форм власності [5].

Отриманні дані GSM сигналу які приймає спеціалізована GSM-антена, що здатна сприймати відповідний сигнал, передаються на GSM-приймач, а саме електронний прилад, який перетворює радіосигнали в цифрові дані використовуючи ключі для розшифровки. Приймач GSM зазвичай складається з фільтру який виконує видалення небажаних сигналів з радіосигналу отриманого антеною, підсилювачу для посилення радіосигналу, демодулятора який виконує основну роль для перетворення радіосигналу на цифровий сигнал.

Зміст пакету з даними GSM-ефіру у цифровому вигляді являє собою послідовність байтів, які представляють собою сигнали, що надходять з базових станцій та мобільних телефонів. Ці сигнали дають можливість бути використаними для моніторингу мережі GSM, а також для виявлення та розслідування злочинів, пов'язаних із мобільним зв'язком.

Пакет з даними GSM-ефіру може бути відкритий за допомогою програмного забезпечення, такого як аналізатор спектра або генератор графіків. Це програмне забезпечення може використовуватися для перегляду сигналів, що містяться у файлі, а також для їх аналізу.

Дані, які генерує демодулятор (пакет даних GSM), містить різноманітні дані, а саме:

- Тип пакету;
- Номер каналу (наприклад голосовий канал);
- Номер абонента;
- Довжина пакета;
- Дані

- код мережі, до якої підключено базову станцію;
- код базової станції;
- потужність сигналу базової станції;
- координата X базової станції;
- координата Y базової станції;

Саме інформація про координати відносно базової станції відповідає за місце знаходження абонента який взятий до моніторингу. Для дешифрування даних використовується таблиця для розшифрування даних GSM. Таблиця GSM є набір символів, кожен із яких відповідає конкретному набору бітів.

Варто зазначити що не маючи кодів доступу декодування блоків і отримання коректних даних близиться до нуля. Метод шифрування не залежить від типу переданих даних (мова, дані користувача або повідомлення сигналізації). Воно застосовується лише до нормальних пакетів (normal burst).

Шифрування досягається перетворенням за допомогою операції "що виключає АБО". Ця операція проводиться між псевдовипадковою багаторозрядною послідовністю та корисними бітами нормального пакета (усі інформаційні біти крім 2 контрольних прапорів). Псевдовипадкова послідовність виходить з урахуванням номера пакета і сеансового ключа (session key). Ключ встановлюється на початку сеансу шляхом обміну сигналами між приймачем та передавачем, використовується лише протягом одного сеансу зв'язку. Після закінчення сеансу ключ автоматично стирається, а розшифровка використовує той самий ключ шифрування. [6]

Також для виявлення місцезнаходження та виявлення цілі, яка випромінює GSM сигнал, можна використовувати метод триангуляційного аналізу місцевості. Маючи відповідне обладнання, відстежувати об'єкт можна не маючи ключі розшифровки сигналу, а безпосередньо за GSM випромінюванням.

Перевага триангуляльного методу полягає в простоті його технічної реалізації, тому він широко застосовується. Істотним недоліком методу є наявність помилкового виявлення неіснуючих джерел при великій кількості випромінюючих об'єктів в зоні покриття пеленгатора. [7]

Ідея триангуляції полягає в тому, що на випромінюючий об'єкт визначається напрямок з різних точок простору, після чого дальність до об'єкта визначається відомими кутами та відстанями між содарною станцією (рис. 1.6).

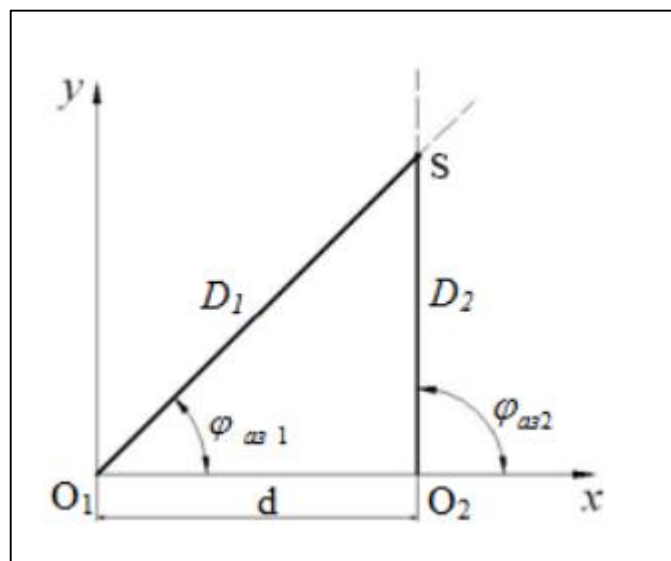


Рисунок 1.6 – Триангуляційний метод позиціонування

(Джерело: <https://er.nau.edu.ua/bitstream/NAU/53699/1/Кваліфікаційна%20робота%20Голдовский%20М..pdf>)

Якщо джерело радіовипромінювання розташоване в горизонтальній або вертикальній площині, то для визначення його розташування достатньо виміряти два азимутальних кута: $\varphi_{аз1}$ і $\varphi_{аз2}$, а розташування джерела випромінювання визначається точкою перетину $O1S$ і $O2S$.

На вимірююму пеленгація кути шляхом простих обчислень, використовуючи теорему синуса, можна визначити діапазони від точок прийому до джерела радіовипромінювання.

$$D_1 = \frac{d \sin(180 - \varphi_{аз2})}{\sin(\varphi_{аз2} - \varphi_{аз1})}, \quad (1.1)$$

$$D_2 = \frac{d \sin \varphi_{a31}}{\sin(\varphi_{a32} - \varphi_{a31})}, (1.2)$$

Перевага триангуляльного методу полягає в простоті його технічної реалізації, тому він широко застосовується.

Приймачами GSM сигналу для збору даних можуть виступати вже існуючі прилади які представленні на ринку, а саме аналізатори спектру (рис. 1.7). Саме такі прилади можуть отримувати значення спектру у певній смузі частот, яка може бути змінною для кожного приладу у різний час.



Рисунок 1.7 – Аналізатор спектра

(Джерело: <https://inlab.kyiv.ua>)

Існуючі аналізатори спектра, в основному мають швидкість прийому даних від приладу приблизно від 20 до 200 спектрограм на секунду. Отримання

даних від приладу надаються в залежності від координати та вектору напрямлення приймача

1.2 Аналіз існуючих СШ й інтелектуальних систем предметної області

СШ та інтелектуальні системи у даній предметній області можуть використовуватися для покращення ефективності моніторингу GSM сигналу. СШ можуть використовуватися для автоматизації процесу збору даних з пристроїв GSM моніторингу, а інтелектуальні системи в свою чергу можуть використовуватися для аналізу отриманих даних, з метою подальшого використання.

СШ для автоматизації процесу збору даних можуть використовуватися для розміщення GSM-сканерів у певних точках і для автоматичного збору даних з цих сканерів. Це може звільнити персонал від необхідності вручну встановлювати та налаштовувати GSM-сканери, а також від необхідності регулярно збирати дані з них.

Інтелектуальні системи можуть використовуватися для аналізу даних, отриманих з GSM-сканерів, що може допомогти в прогнозуванні, шляху переміщення жжерела випромінювання при певних погодних умовах, що покращує ефективність моніторингу GSM сигналу.

1.3 Постановка проблеми та формування задач

Постановка задачі для розробленої експертної системи полягає в тому, щоб на основі отриманих даних які надходять з зовнішніх датчиків моніторингу, а саме з GSM антени, яка генерує інформацію щодо сили наявного сигналу та

його спектру, та даних з генерації отриманих координат випромінюваного сигналу, робити аналіз даних та передбачати теоритичне пересування ворожих військових сил, для подальшого планування з їхнього перехоплення або з їхньої ліквідації. Поставленна задача полягає в тому, щоб забезпечити зручний спосіб моніторингу та обробки інформації, в реальному часі, що дозволить ефективно творити опір стосовно ворожих сил у різний час доби та при будь-яких погодних умовах.

Експертна система призначена для допомоги військовим на полі бою, приймати рішення щодо наступу або перехоплення ворожих сил з метою їх подальшою ліквідацією. Також дана експертна система має місте з використанні у штабах для аналізу інформації, що дозвоилить проробляти різні стратегії на полі бою.

Основна задача еспертної системи полягає в тому, щоб зробити моніторинг з наявності ворожих сил, та отримання їхнього місцеположення, після чого зберігти, та проаналізувати дані. Виконання всіх дії та операцій задачі експертної системи дозволять допомогти військовому складу збільшити вірогідність збереження життів, та якомого максимально оптимізувати всі види ресурсів таким чином, що дозволить ефективно чинити військовий опір щодо ворожиж сил. Для досягнення цієї мети експертна система повинна отримувати дані моніторингу, зберігати велику кількість отриманих даних у сховищах даних, робити аналіз даних щодо пересування ворожих сил, виводити інформацію в зручний, та інтуїтивно зрозумілий інтерфейс для котистувачів. Всі отримані дані, та доступ до інтерфейсу повинні шифруватися для забезпечення високого рівня безпеки. Експертна система повинна мати можливість виключення доступу для військових підрозділів які не вийшли на зв'язок зі штабом, з метою виключення можливості використання системи з боку ворожої сторони.

Задачі моніторингової експертної системи включають:

Збір та аналіз даних наявної присутності ворожих сил, та їхне місцеположення;

Зберігання великих обсягів даних у шифрованому вигляді, для подальшого їх використання та обробки;

Аналіз та розрахунок координат теоретичного пересування ворожих сил;

Вивід інформації в користувачський інтерфейс, для використання отриманого результату військовими силами;

Можливість керування правами доступу до експертної системи з метою запобігання витоку даних.

Експертна система повинна працювати в автономному режимі, та у режимі онлайн, також система повинна мати зв'язок з сховищем даних для подальшої взаємодії з ним. Також система повинна передбачати шифрування даних, як для даних які зберігаються у сховищі даних, так і шифрування інформації для авторизації і аунтифікації користувачів експертної системи.

1.4 Обґрунтування вибору підходів і технологій для проектування та створення експертної системи та її компонентів

У проектуванні експертної системи поєднані різні технології та підходи, які дозволяють робити аналіз спектру та повідомляти про наявність пристроїв, які випромінюють шуканий сигнал у відповідному до того діапазону частот. Передбачено використання технології, які дозволяють керувати пристроєм через WEB-застосунок.

Технологія Java - це комп'ютерна платформа та мова програмування, які розроблені компанією Sun Microsystems (тепер частина Oracle Corporation). Вона включає в себе різноманітні інструменти та бібліотеки, які дозволяють розробникам створювати крос-платформені та надійні програми.

Spring Boot - це фреймворк для розробки Java-додатків, який спрощує створення стійких, ефективних та гнучких додатків. Він є частиною екосистеми

Spring, і його основна мета - забезпечити швидку розробку додатків з мінімальними зусиллями на конфігурацію та підтримку.

Для написання прошивки до мікроконтролерів STM32 використовуються різні інструменти та технології, зазвичай засновані на мові програмування C/C++. Для відлагоджування прошивки на STM32 використовуються різні засоби, такі як ST-Link та внутрішні відлагоджувачі мікроконтролера.

PostgreSQL - це відкрита реляційна база даних, яка зарекомендувала себе як надійний та гнучкий інструмент для зберігання та управління даними. PostgreSQL використовує SQL, стандартизовану мову запитів, що робить її доступною та зручною для користувачів різного рівня підготовки. PostgreSQL гарантує цілісність даних за допомогою транзакцій, забезпечуючи атомарність та послідовність операцій з даними.

3D моделювання - формування цифрових тривимірних моделей, які з максимальною точністю демонструють форму, розміри, зовнішній вигляд та інші характеристики об'єкту. Технології 3D моделювання дозволяють уявити деталь або готовий виріб з усіх ракурсів та внести необхідні виправлення ще на етапі створення. Для 3D візуалізації об'єктів використовується комп'ютерна програма – SolidWorks, в якому формати моделей сумісні з сучасними засобами виробництва: зокрема, 3D друком. Тому після завершення моделювання можна створити прототипи, а далі — запустити виробництво. 3D моделювання прискорює процес розробки, робить його економічно вигіднішим і зручнішим.

SolidWorks - это программа для твердотельного моделирования (CAD) и автоматизированного проектирования (CAE).

2 АНАЛІТИЧНИЙ РОЗДІЛ. ХАРАКТЕРИСТИКА ЕКСПЕРТНОЇ СИСТЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

Аналітичний розділ включає в себе характеристику експертної системи, яка оснований на характеристичних відомостях об'єкта дослідження що дає змогу зробити коректну постановку задачі. Правильно поставлена задача з реалізації експертної системи допомагає збалансувати фізичні і фінансові ресурси, та знайти оптимальне рішення для ринку, яке буде здатне покрити потреби для певних напрямків які передбачають моніторинг сигналів.

2.1 Характеристика об'єкта дослідження

Об'єктом дослідження експертної системи виступає GSM антена (Аналізатор спектра), а саме антена “TinySA Ultra”.

Даний аналізатор спектра виявляє та повідомляє оператора про наявність безпілотних літальних апаратів та інших пристроїв, які випромінюють сигнали у діапазоні частот від 35 Гц до 6 ГГц [8]. Саме цей діапазон включає більшість дронів, в тому числі Lancet, Orlan, Zala, FPV, DJI GNSS service, Wi-Fi, радіо та мобільні сервіси, DCS, GSM, GPRS, ГЛОНАСС, тощо.

Антенa “TinySA Ultra” має здатність улавлювати діапазон частот: від 35 Гц до 6 ГГц, на максимальну дальність виявлення 3.5 - 4.5км. Антенa у повній комплектації важить 2.5 кг, що вважається оптимальним навантаженням на встановлення антени на штатив та рухомі елементи.

Методами оповіщення антени в залежності від необхідного, можуть виступати як повідомлення на дисплей, так і повідомлення через звуковий сигнал (при наявності навушників або колонок). Антенa має можливість взаємодіяти з зовнішніми пристроями такими як комп'ютер або ноутбук.

Антенa “TinySA Ultra” має захищений корпус від пилу та бризг, який включає в себе сенсорний дисплей для керування та налаштування пристрою.

Базова автономність роботи аналізатора спектра складає 2-4 години. Час повного заряджання аналізатора спектра становить 1 годину. При цьому експлуатаційна температура приладу становить від -25 до +50°C, а в умовах вологості до 80% до +25°C.

2.2 Постановка задачі

Задача з реалізації концепції експертної системи полягає у інтеграції GSM антени з окремо створеним програмним додатком, який дозволить повноцінно взаємодіяти як і з функціоналом антени, так і з масивом даних який генерується під час її використання. Окремо розроблений додаток має мати користувальський інтерфейс для зручного користування, а саме форми авторизації у систем, графічний інтерфейс з можливістю встановлення необхідних налаштувань антени та інтерфейс моніторингу з пересування зафіксованих об'єктів які випромінюють певну частоту.

Інтерфейс для авторизації користувачів системи має відповідати вимогам шифрованої передачі даних, та мати доступ до бази даних де зберігаються дані для авторизації у зашифрованому вигляді.

Графічний інтерфейс налаштувань повинен передбачати можливість встановлення налаштувань для режиму роботи антени.

В інтерфейсі радару моніторинга повинні відображатися виявлені об'єкти які зафіксувала антена, для визначення поточних координат тіла, та розрахунки теоретичних координат майбутнього положення тіла у просторі на основі штучного інтелекту.

Експертна система має містити електронні модулі для керування кроковими двигунами, керуючий модуль у вигляді мікроконтролера, засоби живлення та засоби обміну інформації мікроконтролера з комп'ютером.

3 КОНСТРУКТИВНИЙ РОЗДІЛ. РОЗРОБЛЕННЯ ПРОЕКТНИХ РІШЕНЬ

Проектне рішення яке здатне реалізувати експертну систему, яка здатна допогти державі у вирішенні військових питань, як і з точки зору захисту, так і оптимізації ведення наступальних операцій. На основі отриманих даних з зовнішніх датчиків моніторингу, експертна система може дати перевагу для зброїних сил, та дати можливість оптимізувати стратегію. Розділ розкриває в собі дослідження зі створення 3D моделі приладу, побудови бази даних, програмні та апаратні рішення які використовуються в проєкті, програмні інструменти та середовища які використовуються в процесі розробки експертної системи.

3.1 Моделювання та проектування 3D моделі для експертної системи

Експертна система передбачає реалізацію 3D моделі яка повинна тримати GSM антену та її компоненти у точці простора. Така 3D модель реалізовує в собі штатив (рис. 3.1) який повине легко складатись та розкладатись, а також мати можливість легкого монтажу всіх необхідних компонентів. Виготовлятися даний штатив повинен з легкого металу, який має захист від корозії та впливу іншого характеру що може нанести погода та природні явища.



Рисунок 3.1 – Штатив

(Джерело: розроблено автором)

Реалізований штатив має в собі різноманітні деталі, вузли та механізми. Одним з таких вузлів виступає збірка опори (триноги), яка дозволяє зафіксувати фізично прилади експертної системи. Кожна з опор рівномірно фіксується в тілі – триноги (рис. 3.2) за допомогою втулок, на рівномірній між собою відстані, що дозволяє збалансувати навантаження на деталі та механізми, а ще зменшити вірогідність опрокинення конструкції.

3D модель штативу на основі відповідних 2D креслень створено в програмному середовищі SolidWorks, що дозволяє не тільки віддати креслення на виготовлення деталі на виготовлення, але й проконтролювати всі взаємодіючі механізми в яких заздалегіть встановлені необхідні ступені свободи.

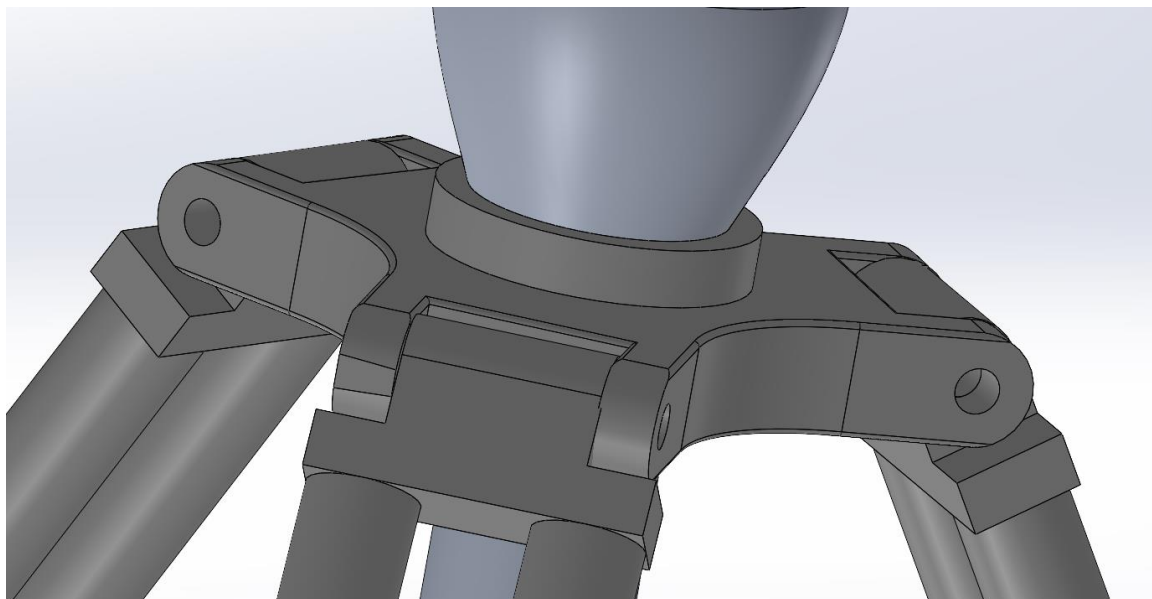


Рисунок 3.2 – Тіло – триноги

(Джерело: розроблено автором)

В центральний отвір тіла триноги фіксується голова штативу на яку фіксується платформа з обладнанням за допомогою рейки. У ролі рейки виступає деталь зі спеціальними пазами типу «ластівчин хвіст». [9] Паз такого типу служить напрямним елементом рухомого вузла платформи (рис. 3.3). Фіксація рейки до платформи відбувається за допомогою гвинта 3/4 дюйма. [10]

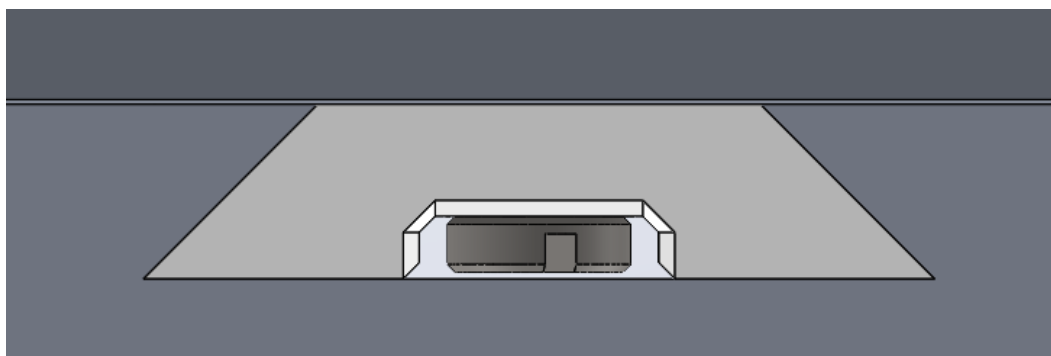


Рисунок 3.3 – Ластівчин хвіст

(Джерело: розроблено автором)

Реалізована модель штативу може бути доопрацьована під різні види та типи обладнання які можуть бути передбачені при подальших модефікаціях всієї концепції.

3.2 Моделювання та проектування бази даних для системи прийняття рішень

У даному проєкті передбачено можливість автоматичного створення необхідних таблиць (у разі їх відсутності) у базі даних, за допомогою Java Spring фреймворку, а саме анотації «@Entity» яка використовується для позначення класу, який представляє сутність або таблицю в базі даних. При використанні анотації «@Entity», клас стає персистентним, тобто його об'єкти можуть бути збережені та отримані з бази даних. [11]

На рисунку 3.4 зображено фрагмент класу який описує таблицю для даних авторизації, яка містить інформацію стосовно користувачів які знаходяться у системі.

```

17
18 @Data
19 @Entity
20 //@Table(name = "users")
21 public class Users{
22     @Id
23     @GeneratedValue(strategy= GenerationType.IDENTITY)
24     private Long id;
25
26     // @Column(name = "email")
27     private String email;
28
29     private String activationCode;
30
31     // @Column(name = "password")
32     private String password;
33     // @Column(name = "first_name")
34     private String firstName;
35     // @Column(name = "last_name")
36     private String lastName;
37
38     @Enumerated(value = EnumType.STRING)
39     // @Column(name = "role")
40     @Column(columnDefinition = "varchar(20) default 'USER'", nullable = false )
41     private ERole role ;
42
43     @Enumerated(value = EnumType.STRING)
44     // @Column(name = "status")
45     @Column(columnDefinition = "varchar(20) default 'ACTIVE'", nullable = false )
46     private EStatus status;
47     public Users () {}
48
49     public Users (String email, String password, String firstName, String lastName, ERole role, EStatus status ) {
50         this.email = email;
51         this.password = password;
52         this.firstName = firstName;
53         this.lastName = lastName;
54         this.role = role;
55         this.status = status;
56     }
57
58     public Users (String email) {
59         this.email = email;
60     }
61 }

```

Рисунок 3.4 – приклад класу який описує таблицю користувачів

Джерело: розроблено автором

Даний клас взаємодіє з допоміжною таблицею в якій описані ролі користувачів, що дозволяє кожному окремому користувачу давати певні права. Даний підхід дозволить розвивати функціонал експертної системи у майбутньому та підвищити рівень безпеки програмного застосунку. Фрагмент класу для генерації таблиці з ролями зображено на рисунку 3.5.

```

6
7  /**
8   *
9   * @author Dima
10  */
11  import org.springframework.security.core.authority.SimpleGrantedAuthority;
12  import java.util.Set;
13  import java.util.stream.Collectors;
14
15  public enum ERole {
16
17      USER(permissions: Set.of(e1: Permission.DEVELOPERS_READ)),
18      ADMIN(permissions: Set.of(e1: Permission.DEVELOPERS_READ, e2: Permission.DEVELOPERS_WRITE)),
19      OWNER(permissions: Set.of(e1: Permission.DEVELOPERS_READ, e2: Permission.DEVELOPERS_WRITE, e3: Permission.DEVELOPERS_CREATE));
20
21      private final Set<Permission> permissions;
22
23      ERole(Set<Permission> permissions) {
24          this.permissions = permissions;
25      }
26
27      public Set<Permission> getPermissions() {
28          return permissions;
29      }
30
31      public Set<SimpleGrantedAuthority> getAuthorities() {
32          return getPermissions().stream()
33              .map(permission -> new SimpleGrantedAuthority(role: permission.getPermission()))
34              .collect(collector:Collectors.toSet());
35      }
36
37  }
38

```

Рисунок 3.5 – приклад класу який описує таблицю ролів

Джерело: розроблено автором

Аналогічно створюються усі інші таблиці, які фігурують у проекті. Структури інформаційних масивів описані в таблицях 3.1 – 3.2.

Таблиця 3.1

Найменування	Ідентифікатор у програмі	Ум. Позн. Форм.	Формат	PK\FK	Бізнес-правила		
					Умова на значення	Обов'язкове поле	Індексне поле
Ідентифікатор	ID_gsm	Gsm	Int	PK	Not null	Так	ІНД

Дата та час запису	date_and_time_of_recording	Da	Varchar	-	Not null	Так	-
частота вимірювання	measurement_frequency	Me	Int	-	Not null	Так	-
Вимірний рівень потужності сигналу	measured_signal_strength_level	Si	Int	-	Not null	Так	-
Ширина смуги сигналу	Signal_band_width	Ba	Int	-	Not null	Так	-
Тип сигналу	signal_type	St	Varchar	-	Not null	Так	-
Ідентифікатор базової станції	base_station_identifier	Bs	Varchar	-	Not null	Так	-
Тип шифрування	Encryption_type	Et	Varchar	-	Not null	Так	-
Код місцевості	area_code	Ac	Varchar	-	Not null	Так	-
Рівень якості прийнятого сигналу	Quality_level_of_the_received-signal	Ls	Varchar	-	Not null	Так	-
Номер каналу GSM	gsm_channel_number	Nc	Int	-	Not null	Так	-

Джерело: сформовано автором на основі виконаного дослідження

Таблиця 3.2

Найменування	Ідентифікатор у програмі	Ум. Позн. Форм.	Формат	PK\FK	Бізнес-правила		
					Умова на значення	Обов'язкове поле	Індексне поле
Ідентифікатор	ID_user	Us	Int	PK	Not null	Так	ІНД
Пошта	email	Em	Varchar	-	Not null	Так	-
Ім'я	first_name	Fn	Varchar	-	Not null	Так	-
Прізвище	Last_name	Ln	Varchar	-	Not null	Так	-
пароль	password	Pw	Varchar	-	Not null	Так	-
Роль	role	Ro	Varchar	-	Not null	Так	-
Статус	status	St	Varchar	-	Not null	Так	-

Джерело: сформовано автором на основі виконаного дослідження

Уся інформація яка описує дані для з'єднання застосунку з базою даних, знаходиться у файлі «application.properties» (рис. 3.6), а саме логін, пароль, порт для підключення до бази даних, та назва бази даних.

```

13
14 spring.jpa.hibernate.ddl-auto=update
15 spring.datasource.url=jdbc:mysql://${MYSQL_HOST:localhost}:3306/springweb?useTimezone=true&serverTimezone=UTC
16 spring.datasource.username=root
17 spring.datasource.password=root
18

```

Рисунок 3.6 – З'єднання застосунку з базою даних

Джерело: розроблено автором

Для організації збору, і передавання інформації у програмній реалізації виступають так звані класи-репозиторії у яких описуються запити до бази даних. Для прикладу розглянемо код, який представляє метод, який оновлює статус та роль користувача в базі даних, з використанням умови фільтрації за іменем користувача. (рис. 3.7)

```

74     @Modifying
75     @Query("update Users u set u.role = :role, u.status = :status where u.email = :email")
76     void updateInformation(@Param(value = "email") String email,
77         @Param(value = "role") ERole role,
78         @Param(value = "status") EStatus status);
79 }

```

Рисунок 3.7 – Запити до бази даних

Джерело: розроблено автором

Структура даного функціоналу містить як запит, який оновлює поле role та поле status у сутності Users залежно від заданих умов, так і необхідні анотації. Анотація «@Modifying» яка вказує, що метод виконує модифікуючий запит до

бази даних (оновлення, видалення або вставка). Анотація «@Query» використовується для визначення власного JPQL запити. Анотація «@Param» використовується для мапування параметрів методу на іменовані параметри запити. Атрибут «value» вказує назву параметра у запиті. Функціонал того чи іншого запити до бази даних виконується шляхом його виклику.

3.3 Розроблення користувацького інтерфейсу. Елементи та структура

Користувальський інтерфейс для розробленої системи являє собою розроблене середовище в якому відображаються координати знайдених джерел випромінювання (рис. 3.8) та їхні розраховані теоретичні координати у майбутньому часі які розраховуються на основі штучного інтелекту. Координати розраховуються відносно встановленого приладу – аналізатора GSM сигналу.

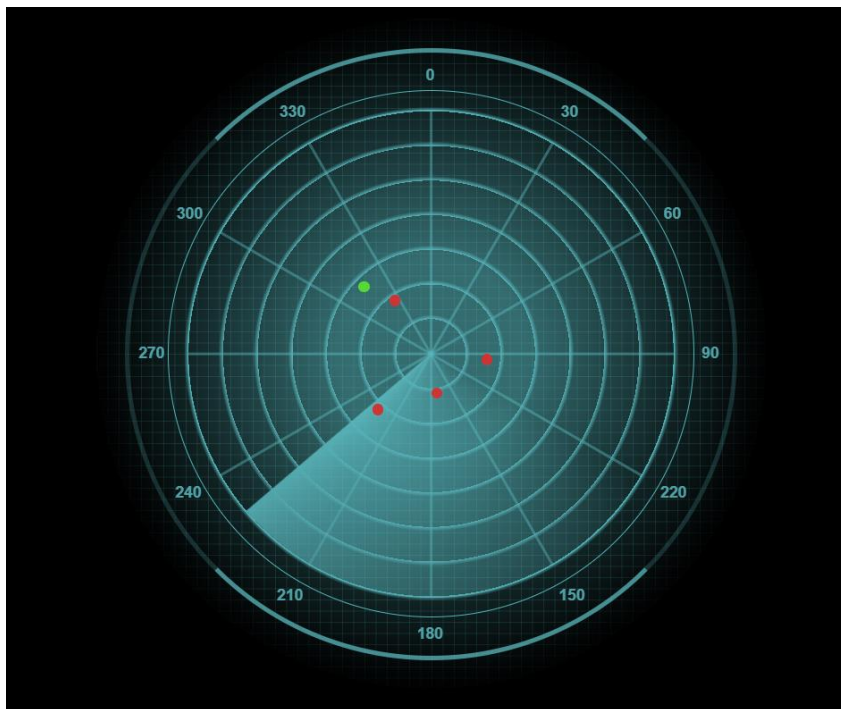
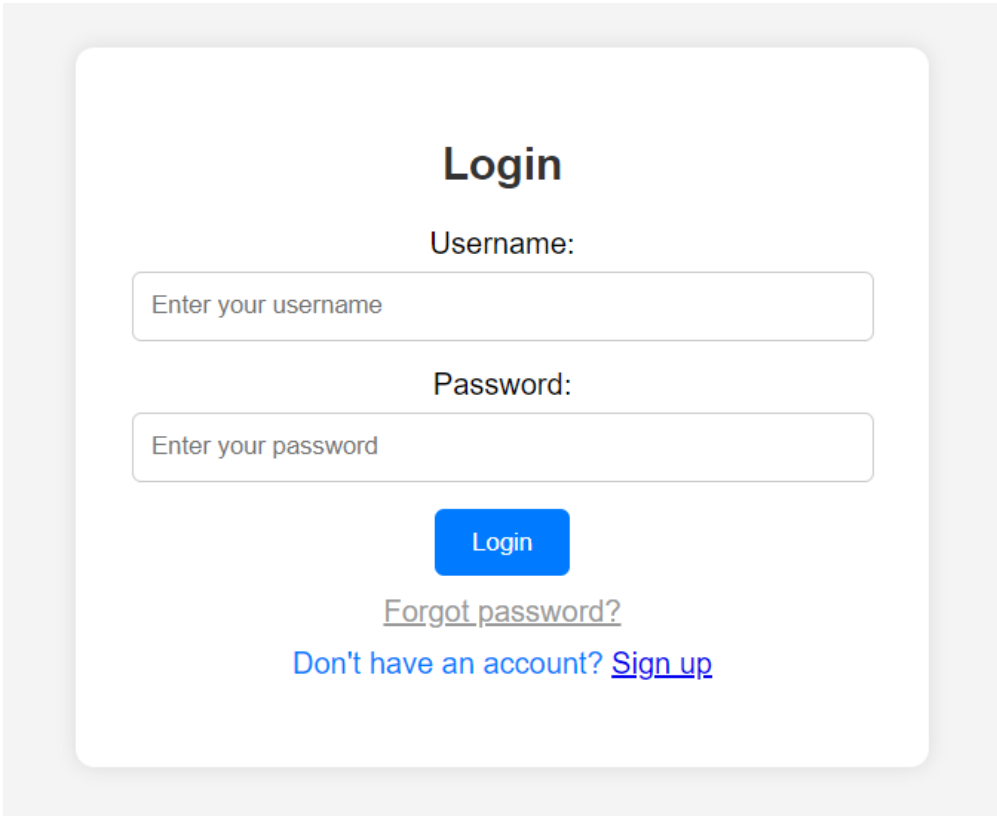


Рисунок 3.8 – Користувальський інтерфейс радару

Джерело: розроблено автором

Для входу у систему використовується відповідна форма (рис. 3.9), яка дозволяє авторизуватись користувачу та почати використовувати прилад. Авторизація передбачає передачу логіну та паролю користувача на сервер в зашифрованому вигляді, де подалі з отриманих даних генерується хеш код та порівнюється з хеш-кодом який зберігається в базі даних.



The image shows a login form titled "Login". It contains two input fields: "Username:" with the placeholder text "Enter your username" and "Password:" with the placeholder text "Enter your password". Below the password field is a blue "Login" button. Underneath the button are two links: "Forgot password?" and "Don't have an account? Sign up".

Рисунок 3.9 – Форма входу

Джерело: розроблено автором

Користувач системи має змогу обрати необхідні для аналізу діапазони частот в інтерфейсі (рис.3.10), на основі яких і буде проводитись аналіз даних. Фільтрація частот дозволить виділити необхідні джерела випромінювання для подальшої роботи з ними.



Рисунок 3.10 – інтерфейс налаштувань

Джерело: розроблено автором

3.4 Моделювання та проектування апаратної реалізації для експертної системи

Експертна система має спроектований апаратний модуль для керування кроковими двигунами для забезпечення можливості змінювати направленість встановленої GSM антени.

Для керування кроковим двигуном за допомогою мікроконтролера використовується апаратна реалізація, яка дозволяє подати на котушки необхідну напругу. Дана апаратна реалізація посилює сигнал з мікроконтролера та керує полярністю задіяної котушки крокового двигуна.

Апаратна реалізація складається з шести транзисторів, чотири з яких є силовими, а два керуючими (рис. 3.11). Дана система дозволяє пропускати через двигун струм в обидві сторони і регулювати його потужність. У реалізації даної схеми використовуються транзистори р-п-р типу (КТ814) та n-р-п (КТ961). Резистори використовуються по 4,7 КОМ і по 10 КОМ.

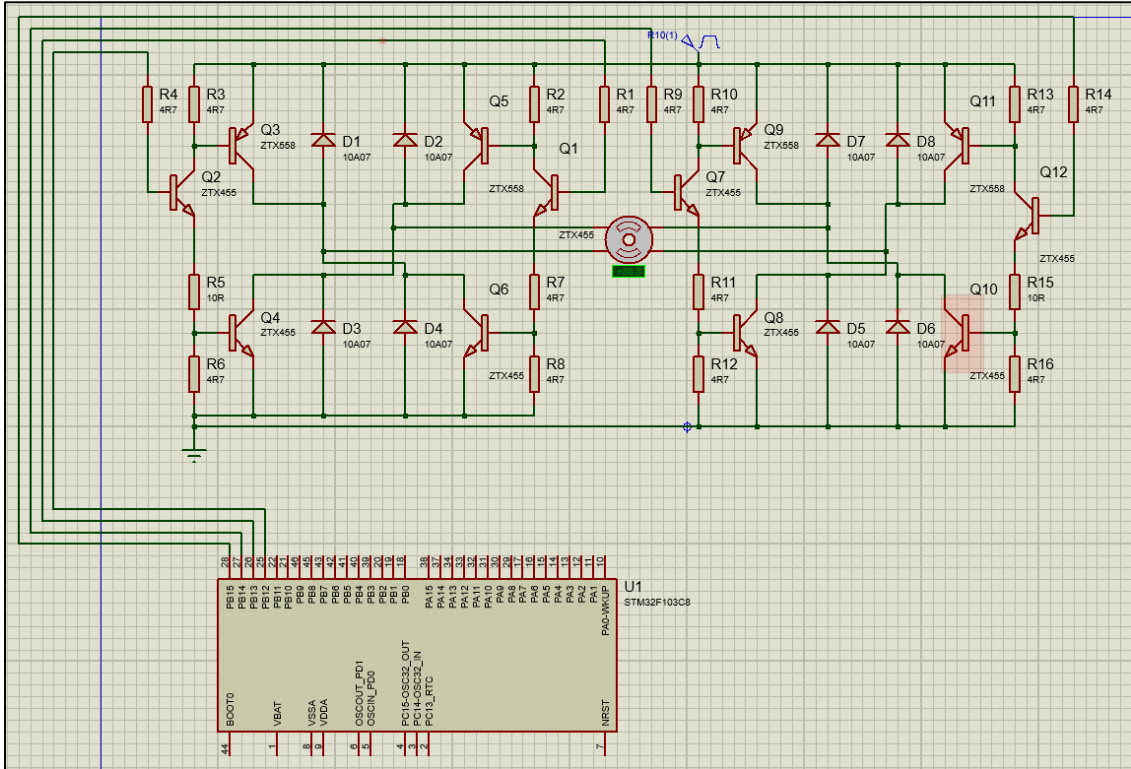


Рисунок 3.11 – Апаратна реалізація

Джерело: розроблено автором

Працює апаратна реалізація шляхом отримання сигналів, на логічні входи (рис. 3.12). При зміні шпаруватості поданого сигналу можна керувати швидкістю обертання кроковим двигуном [12].

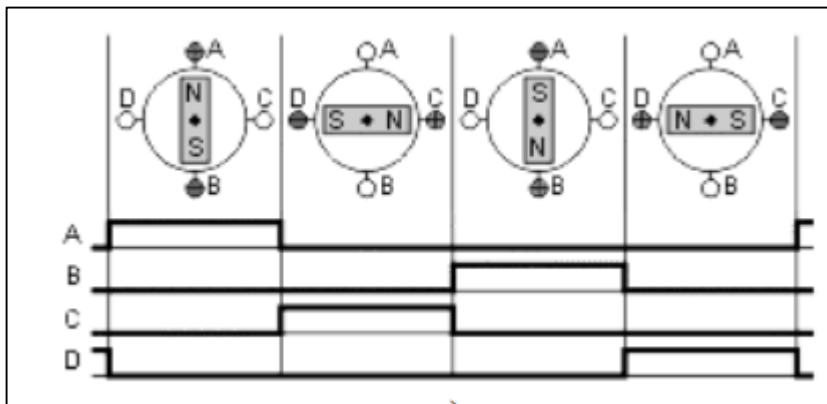


Рисунок 3.12 – Апаратна реалізація

Джерело: розроблено автором

Для зв'язку мікроконтролера з комп'ютером використовується інтерфейс передачі даних - UART, який реалізує обмін даними по двох дротах між двома пристроями, при якому передача та прийом даних відбувається по двох роздільних дротах. Причому контакт передачі даних одного пристрою з'єднується з контактом прийому даних іншого і навпаки (рис. 3.13). Для цього в середовищі «CubeMX» у пункті меню Connectivity вмикаємо UART1 в положення Asynchronous, після чого у налаштуваннях в Configuration встановлюємо параметри які представлено на рисунку 3.14.

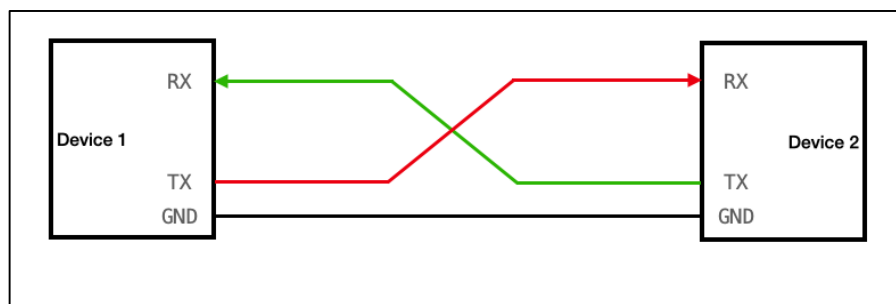


Рисунок 3.13 – Схема з'єднання контактів передачі даних

Джерело: розроблено автором

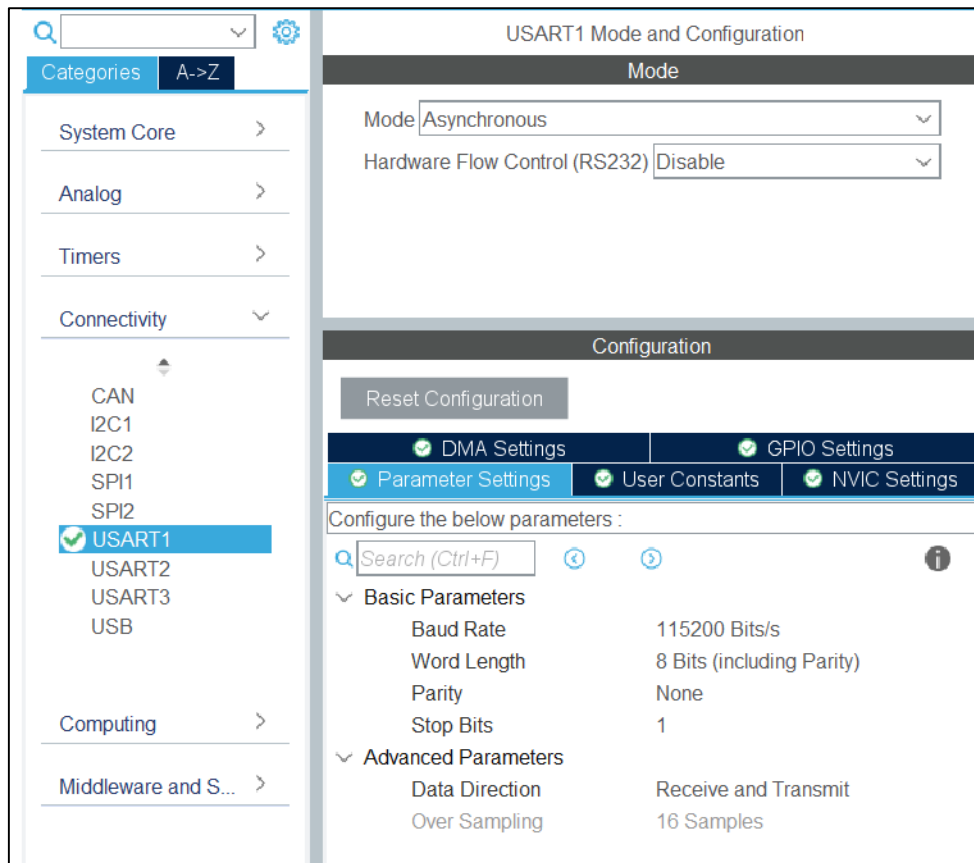


Рисунок 3.14 – Встановлення параметрів

Джерело: розроблено автором

У реальному часі відбувається прослуховування інтерфейсу UART, та в залежності від отриманого коду (даних), програма продовжує керуватись своїм алгоритмом [13]. При отриманні відповідного коду по UART програма виконує відповідні дії, а саме перевіряє логічний стан ініціалізації антени, передає інформацію про стан крокових двигунів, рівень сигналу з антени, тощо. Отримана інформація про стан через UART інтерфейс відправляється на сервер. При отриманні відповідних керуючих сигналів на мікроконтролер встановлюється відповідний режим роботи всієї системи.

3.5 Проектування забезпечувальних підсистем експертної системи

Забезпечувальні підсистемами експертної системи складаються з розробленого програмного забезпечення, яке забезпечує коректну та надійну роботу всієї системи. Програмна складова забезпечувальних підсистем грає основну роль в коректному функціонуванні всіх складових модулів експертної системи.

Реалізація класу для здійснення обміну інформації з МК через UART інтерфейс використовує бібліотеку «jSerialComm», яка являє собою Java-бібліотеку, що надає зручний інтерфейс для роботи з послідовними (UART/COM) портами на різних платформах.

Взаємодія з послідовним портом відбувається шляхом пошуку необхідного порту, після чого відбувається встановлення параметрів конфігурацій порту, такі як швидкість передачі, біти даних, стоп-біти та парність.

Швидкість передачі (Baud Rate): Швидкість передачі в бодах вказує, як швидко дані будуть передаватися через порт. Встановлюється за допомогою методу «setBaudRate(int baudRate)».

Біти даних (Data Bits): Кількість бітів даних в одному байті. Зазвичай це 8 бітів. Встановлюється за допомогою методу «setNumDataBits(int dataBits)».

Стоп-біти (Stop Bits): Кількість стоп-бітів, які додаються після кожного байта для сигналізації завершення передачі. Зазвичай це 1 стоп-біт. Встановлюється за допомогою методу «setNumStopBits(int stopBits)».

Реалізований клас має методи які дозволяють відкрити порт, відправити або отримати дані і закриває порт (рис. 3.15, рис. 3.16).

```

public void writeLine(String data) throws IOException {

    byte[] b = data.getBytes();
    int l =
    activePort.writeBytes(buffer:b, bytesToWrite: b.length);
    if (l == -1) {
        throw new IOException(message: "Write error.");
    }
}

public void closePort() {

    activePort.closePort();
}

```

Рисунок 3.15 – Програмні методи для відправки даних та зачинення порту

Джерело: розроблено автором

```

@Override
public void serialEvent(SerialPortEvent event) {

    int size = event.getSerialPort().bytesAvailable();
    byte[] buffer = new byte[size];
    event.getSerialPort().readBytes(buffer, bytesToRead: size);
    String str="";
    for(byte b:buffer){
        System.out.print((char)b);

    str = str + (char) b;
    if(str.equals(anObject: "comand: 001")){
        System.out.println(x: "comand: 001");
    }
    }
    System.out.println(x: "#closePort");
    closePort();
}

```

Рисунок 3.16 – Програмний метод для отримання даних

Джерело: розроблено автором

Реалізація забезпечувальної підсистеми на основі ШІ з використанням фреймворку Java Spring, здатна обчислювати координати руху об'єкта який випромінює GSM сигнал на основі записаних координат попереднього руху.

Штучний інтелект аналізує записані координати руху та обчислює нові координати. Програмна реалізація базується на основі Deeplearning4j, яка в свою чергу використовується для створення та навчання нейронної мережі.

Програмна реалізація (рис. 3.17) складається з сервісу прогнозування руху який використовується для створення, тренування та використання моделі нейронної мережі. Вхідні нейрони відповідають за координати які отримуються з GSM антени, та являються основними даними моніторингу.

Параметри “numInput” і “numOutputs” відповідають за кількість вхідних і вихідних нейронів, які відповідають за координати. Параметр “DenseLayer” відображає кількість нейронів у прихованому шарі. Прихований шар у нейронній мережі має кілька важливих функцій, і кількість нейронів у цьому шарі має значний вплив на здатність мережі вчитися та адаптуватися до складних шаблонів у даних.

Параметр “learningRate” відповідає за швидкість навчання для алгоритму. Конфігурація мережі створюється за допомогою патерну Builder, що дозволяє налаштовувати параметри нейронної мережі.

```

import org.deeplearning4j.nn.conf.layers.DenseLayer;
import org.deeplearning4j.nn.conf.layers.OutputLayer;
import org.deeplearning4j.nn.multilayer.MultiLayerNetwork;
import org.deeplearning4j.nn.weights.WeightInit;
import org.nd4j.linalg.activations.Activation;
import org.nd4j.linalg.api.ndarray.INDArray;
import org.nd4j.linalg.factory.Nd4j;
import org.nd4j.linalg.learning.config.Sgd;
import org.nd4j.linalg.lossfunctions.LossFunctions;
import org.springframework.stereotype.Service;

/**
 * Service for predicting movements based on latitude and longitude inputs.
 * @author Dima
 */
@Service
public class MovementPredictionService {
    private MultiLayerNetwork model;

    public MovementPredictionService() {
        int numInput = 2;
        int numOutputs = 2;
        int numHiddenNodes = 20;
        double learningRate = 0.01;

        MultiLayerConfiguration conf = new NeuralNetConfiguration.Builder()
            .updater(new Sgd(learningRate))
            .weightInit(WeightInit.XAVIER)
            .list()
            .layer(0, new DenseLayer.Builder().nIn(numInput).nOut(numHiddenNodes)
                .activation(Activation.RELU)
                .build())
            .layer(1, new OutputLayer.Builder(LossFunctions.LossFunction.MSE)
                .activation(Activation.IDENTITY)
                .nIn(numHiddenNodes).nOut(numOutputs).build())
            .build();

        this.model = new MultiLayerNetwork(conf);
        this.model.init();
    }
}

```

Рисунок 3.17 – Сервіс прогнозування

Джерело: розроблено автором

3.5.1 Розгортання експертної системи та налаштування параметрів

Учасниками взаємодії з експертною системою являються користувачі які відслідковують GSM сигнал для прийняття рішень щодо подальших дій. Основними учасниками взаємодії з інформаційною системою є адміністратори військових штабів які можуть проводити моніторинг у системі та керувати доступом користувачів у систему;

Користувачі можуть взаємодіяти з системою, а саме:

- автотозуватися;
- переглянути данні моніторингу;
- обрати діапазон GSM сигналу;

Експертна система збору та систематизації даних з мережі пристроїв моніторингу предостовляє можливість працювати з виявленими GSM сигналами у модулі радару, та має вимоги:

- Мати захищати дані шляхом їх шифрування;
- Створювати обліковий запис для кожного користувача;
- Мати систему авторизації та реєстрації учасників;
- Взаємодіяти з хранищем даних;

Експертна система збору та систематизації даних з мережі пристроїв моніторингу, передбачає можливість реєструвати учасників системи, входу до облікового запису, та відновлення доступу до системи у разі втрати даних для аутентифікації. Усі паролі які зберігаються у базі даних шифруються за допомогою BCryptPasswordEncoder. Для відновлення або створення облікового запису у системі використовується підтвердження через електронну пошту та підтвердження адміністраторів

BCryptPasswordEncoder є одним з алгоритмів хешування паролів, що використовується для зберігання паролів у безпечному вигляді в базі даних або іншому сховищі. Він є частиною бібліотеки Spring Security у Java-екосистемі.

BCryptPasswordEncoder використовує адаптивний алгоритм хешування bcrypt, який базується на алгоритмі Blowfish і використовує сіль (salt) для забезпечення безпеки паролів. Цей алгоритм вважається одним з найбільш надійних для хешування паролів, оскільки його обчислювальна складність може бути збільшена з часом, ускладнюючи атаки з використанням перебору.

BCryptPasswordEncoder забезпечує безпечну генерацію хешів паролів та перевірку введеного паролю збереженого хешу. Він автоматично генерує унікальну сіль для кожного паролю, що додає додатковий рівень безпеки до процесу хешування.

При перевірці пароля BCryptPasswordEncoder порівнює введений пароль зі збереженим хешем паролю, використовуючи внутрішній механізм порівняння

всупрт. Якщо пароль вірний, порівняння буде успішним, і користувач буде автентифікований.

BCryptPasswordEncoder є популярним вибором для безпечної обробки паролів у багатьох веб-додатках, забезпечуючи захист від атак зламу баз даних та компрометації паролів.

Можливість змінювати данні у системі дозволяється у відповідності до дозволів для тій чи іншій ролі (рис. 3.18).

```

77     .antMatchers( antPatterns: "/newpassword/**").permitAll()
78
79     // .antMatchers(HttpMethod.GET, "/home/**").hasAuthority(Permission.DEVELOPERS_WRITE.getPermission())
80     // .antMatchers(HttpMethod.GET, "/success").hasAuthority("USER")
81     // .antMatchers(HttpMethod.GET, "/pl/**").hasAuthority(Permission.DEVELOPERS_READ.getPermission())
82     // .antMatchers(HttpMethod.GET, "/success/**").hasAuthority(Permission.DEVELOPERS_READ.getPermission())
83
84     .antMatchers( method: HttpMethod.POST, antPatterns: "/caesarCipher").hasAuthority( authority: Permission.DEVELOPERS_READ.getPermission())//1
85
86     .antMatchers( method: HttpMethod.GET, antPatterns: "/success").hasAuthority( authority: Permission.DEVELOPERS_READ.getPermission())//1
87     .antMatchers( method: HttpMethod.POST, antPatterns: "/success").hasAuthority( authority: Permission.DEVELOPERS_READ.getPermission())//1
88     .antMatchers( method: HttpMethod.GET, antPatterns: "/permissionsManagementPage/**").hasAuthority( authority: Permission.DEVELOPERS_CREATE.getPermission())//1
89     .antMatchers( method: HttpMethod.POST, antPatterns: "/permissionsManagementPage/**").hasAuthority( authority: Permission.DEVELOPERS_CREATE.getPermission())//1
90
91

```

Рисунок 3.18 – Опис доступу

Джерело: розроблено автором

Підключення SSL (Secure Sockets Layer) для експертної система збору та систематизації даних з мережі пристроїв моніторингу є важливим кроком для забезпечення безпеки та захисту комунікації між клієнтами та серверами. Такий сертифікат отримується у постачальників даних сертифікатів. Після отримання сертифікату проводиться встановлення його на сервері, де працює експертної система. Це включає завантаження сертифікату на сервер та налаштування веб-сервера (Nginx) для використання SSL.

У файл «your_domain_name.crt», відразу після тексту основного сертифіката копіюємо вміст файлів (файлу) проміжних сертифікатів (кожен наступний сертифікат вказується з нового рядка). У файлі віртуального хоста Nginx для сайту, у конфігурацію (рис. 3.19) додаємо параметр «ssl_certificate», у якому фігурує файл основного та кореневого сертифікатів, та

«ssl_certificate_key» - файл з ключем, який був створений при генерації CSR і починається тегом «-BEGIN PRIVATE KEY-». Для отримання такого файлу необхідно скопіювати ключ у створений на комп'ютері звичайний текстовий документ і зберегти файл таким чином, щоб його назва закінчувалася на «key».

```
server {  
  
    listen 443;  
    ssl on;  
    ssl_certificate /etc/ssl/your_domain_name.crt; (or .pem)  
    ssl_certificate_key /etc/ssl/your_domain_name.key;  
  
    server_name your.domain.com;  
    access_log /var/log/nginx/nginx.vhost.access.log;  
    error_log /var/log/nginx/nginx.vhost.error.log;  
    location / {  
        root /home/www/public_html/your.domain.com/public/;  
        index index.html;  
    }  
  
}
```

Рисунок 3.19 – Файл віртуального хоста Nginx для сайту

Джерело: <https://ssl.com.ua/wiki/how-to-install-ssl-nginx/>

Експертна система збору та систематизації даних з мережі пристроїв моніторингу, розгортається шляхом проведення відповідних для цього заходів. Спочатку робимо канал зв'язку сервера з комп'ютером (рис. 3.20) на якому відбувалася розробка системи, за допомогою протоколу мережевого рівня SSH, який забезпечує захищену та шифровану комунікацію між двома вузлами через небезпечну мережу, таку як Інтернет. Використовуючи SSH, можна встановити забезпечене з'єднання між клієнтом і сервером, що дозволяє виконувати команди та передавати дані без ризику перехоплення або зміни інформації.

```

Dima@DESKTOP-0[REDACTED] MINGW64 /e/[REDACTED] app/[REDACTED]
$ ssh-keygen.exe
Generating public/private rsa key pair.
Enter file in which to save the key (/c:/Users/Dima/.ssh/id_rsa):
/c:/Users/Dima/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c:/Users/Dima/.ssh/id_rsa
Your public key has been saved in /c:/Users/Dima/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:[REDACTED]ko Dima@DESKTOP-0[REDACTED]
The key's randomart image is:
+----[RSA 38[REDACTED]-----+
  oo=.
  .++.
  = o
  * + .
  . * S + .
  = % . +
  . E B . o .
  = @o o .
  .=*@=o ...
+----[S[REDACTED]-----+

Dima@DESKTOP-0[REDACTED] MINGW64 /e/[REDACTED]
$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3[REDACTED]45L43RGv0ZD[REDACTED]id1j5rwG
Dima@DESKTOP-0H5POQS MINGW64 /e/[REDACTED]
$ scp ~/.ssh/id_rsa.pub root@[REDACTED]
root@[REDACTED] password:
id_rsa.pub

```

Рисунок 3.20 – Канал зв'язку сервера з комп'ютером

Джерело: розроблено автором

3.5.2 Обґрунтування вибору розподіленої архітектури

Централізований архітектурний стиль. Описує підхід до проектування та розгортання систем, де всі компоненти і функціональні модулі системи знаходяться в одному центральному місці або на одному центральному сервері [14]. У цій архітектурі центральний сервер виконує багато завдань, включаючи обробку даних, управління ресурсами і забезпеченням доступу до функцій системи.

Цей стиль було обрано, оскільки він має такі важливі переваги:

- Централізоване управління: Усі рішення та керування приймаються центральним сервером. Він відповідає за координацію та керування всіма функціями системи.
- Одина точка входу: Усі запити та комунікація з системою відбуваються через центральний сервер. Це означає, що всі користувачі та компоненти системи повинні мати доступ до центрального сервера для отримання доступу до функцій системи.
- Загальна база даних: Часто в централізованих РІС використовується одна централізована база даних, до якої мають доступ всі компоненти системи. Це дозволяє спільне використання даних та забезпечує єдину точку керування даними.
- Висока надійність: Централізована архітектура може бути легше контрольована та керована, що дозволяє забезпечити високу надійність та доступність системи.

Дворівнева архітектура клієнт-сервер є однією з базових моделей для розподілених систем. У цій архітектурі система розбивається на два рівні: клієнтський рівень і серверний рівень. Кожен рівень виконує певні функції і взаємодіє з іншим рівнем для забезпечення функціональності та обробки даних.

Основна ідея дворівневої архітектури полягає в тому, що клієнтський рівень відповідає за інтерфейс користувача і взаємодію з користувачем, тоді як серверний рівень забезпечує обробку даних і виконання бізнес-логіки [15].

Клієнтський рівень:

- Включає програми або інтерфейси, що взаємодіють з користувачем.
- Здійснює надсилання запитів до сервера для отримання даних або виконання певних дій.
- Обробляє відображення інформації для користувача.

Серверний рівень:

- Забезпечує обробку запитів, отриманих від клієнтського рівня.
- Здійснює доступ до бази даних або інших ресурсів, необхідних для виконання запитів.
- Містить бізнес-логіку, що визначає правила та операції, які виконуються над даними.

Один із ключових аспектів дворівневої архітектури полягає в тому, що клієнтські програми можуть бути розгорнуті на різних платформах (наприклад, настільні комп'ютери, мобільні пристрої), а сервер може бути централізованим або розподіленим.

Модель тонкого клієнта (Thin Client) є альтернативою традиційним моделям клієнт-серверної архітектури. У цій моделі основна обробка і обчислення відбуваються на сервері, тоді як клієнтські пристрої (тонкі клієнти) виконують лише обмежені функції та передають запити та дані на сервер для обробки.

Модель було обрано, оскільки в неї є наступні переваги:

- Мінімальний обсяг обчислень на клієнтському пристрої: Тонкий клієнт зазвичай не має великої потужності обчислювального процесу або об'єму пам'яті. Він виконує лише основні операції взаємодії з користувачем, такі як відображення графічного інтерфейсу, введення даних тощо.
- Централізоване управління: Усі обчислення та обробка даних здійснюються на сервері. Це означає, що адміністрування, поновлення програмного забезпечення та зберігання даних проводяться на сервері, що полегшує управління та підтримку системи.
- Ефективне використання ресурсів: Модель тонкого клієнта дозволяє ефективно використовувати ресурси, оскільки всі обчислення

відбуваються на сервері. Клієнтські пристрої не вимагають потужного обладнання або великої кількості дискового простору, що знижує вартість утримання та поновлення пристроїв.

Архітектурний стиль для реалізації розподіленої архітектури проекту полягає у проектуванні веб-додатку який визначає загальну структуру та організацію компонентів системи. Для реалізації поставленої мети було обрано декілька архітектурних стилів, які використовуються при проектуванні веб-додатку та системи в цілому, а саме:

- Клієнт-серверна архітектура: Цей стиль передбачає розподіл функціональності між клієнтською та серверною частинами системи [16]. Клієнти взаємодіють з сервером, який надає ресурси та послуги, такі як обробка запитів, збереження даних тощо. Цей стиль дозволяє розділити обов'язки та полегшити масштабованість системи.

- MVC (Model-View-Controller): MVC - це архітектурний шаблон, який розділяє систему на три основні компоненти: модель, представлення та контролер. Модель відповідає за управління даними та бізнес-логікою [17]. Представлення відображає дані користувачеві та обробляє взаємодію з ним. Контролер приймає вхідні дані від користувача та керує виконанням дій. Цей стиль дозволяє розділити компоненти та полегшити їхнє тестування та модифікацію.

- REST (Representational State Transfer): REST - це архітектурний стиль, який базується на принципах, що використовуються в Інтернеті [18]. REST передбачає використання стандартних HTTP-методів, таких як GET, POST, PUT і DELETE, для взаємодії з ресурсами на сервері. Додаток REST складається з ресурсів (наприклад, URL-адреси), які можуть бути створені, зчитані, оновлені або видалені за допомогою відповідних запитів. Цей стиль сприяє легкості в розробці, масштабованості та інтеграції систем.

- Single-Page Application (SPA): SPA - це архітектурний стиль, в якому веб-додаток завантажується один раз, а далі взаємодія з сервером відбувається асинхронно без повторного завантаження сторінок [19]. Весь контент та функціонал управління відображаються на одній сторінці, яка динамічно оновлюється за допомогою JavaScript. Цей стиль дозволяє створити швидкий та зручний для користувача інтерфейс, а також полегшує розробку та підтримку додатка.

Наведені архітектурні стилі, які використовуються при проектуванні, розподіленої експертної системи обґрунтовуються тим що, клієнт-серверна архітектура підходить для експертних систем моніторингу GSM сигналу, оскільки дозволяє централізовано управляти даними та бізнес-логікою. Клієнти (наприклад, мобільні додатки або веб-браузери) здійснюють запити до сервера для отримання інформації про меню, замовлення, резервування столиків та інше. Це спрощує розподілення функцій між клієнтською та серверною сторонами, полегшує масштабування та підтримку системи.

MVC є популярним архітектурним стилем для веб-додатків. Він дозволяє розділити логіку додатку на три основні компоненти: модель (Model), яка представляє дані та бізнес-логіку, представлення (View), яке відображає інформацію користувачу, та контролер (Controller), який обробляє запити користувача та керує взаємодією між моделлю та представленням. Цей підхід дозволяє забезпечити розділення обов'язків та покращити повторне використання коду.

SPA архітектурний стиль підходить для експертних систем моніторингу GSM сигналу, оскільки дозволяє створити зручний та ефективний інтерфейс для клієнтів. SPA завантажується один раз, а подальша взаємодія з сервером відбувається без повторного завантаження сторінок. Це дозволяє забезпечити швидку та зручну навігацію, покращити користувацький досвід та зменшити навантаження на сервер.

Загалом, обрані архітектурні стилі (клієнт-серверна архітектура, MVC, REST, SPA) допоможуть створити експертну систему моніторингу GSM сигналу (рис. 3.21), яка буде ефективно взаємодіяти з клієнтами, матиме централізоване управління та забезпечить швидкість та зручність використання [20].

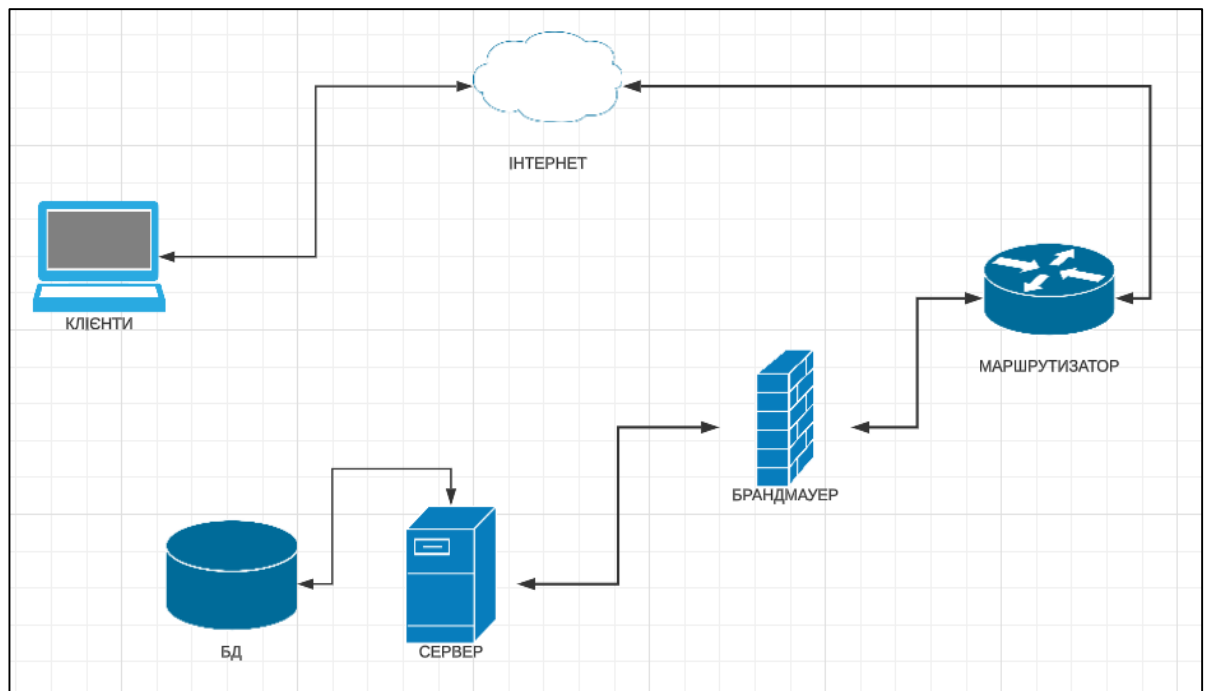


Рисунок 3.21 – Схема розподіленої системи

Джерело: розроблено автором

Необхідні технічні (hardware) вимоги до реалізації обраних типів архітектур представлено в Таблиці 3.3.

Таблиця 3.3

Процесор	Intel Core i5 8700
Відеокарта	Nvidia GTX 1060 Ti
Оперативна пам'ять	4x8 Gb
Накопичувачі HDD	4x1000 Gb RAID 10

Джерело: сформовано автором на основі виконаного дослідження

Необхідні програмні вимоги (software) вимоги до реалізації обраних типів архітектур представлено в Таблиці 3.4.

Таблиця 3.4

Операційна система	Linux Debian 11
Брандмауер	FirewallD
СКБД	PostgreSQL
Комплект розробника	JDK
Проксі-сервер	Nginx
Адміністрування	pgAdmin4
Фреймворк збірки пакетів	Maven
Віддалений доступ	SSH
Середовище розробки	Apache Netbeans

Джерело: сформовано автором на основі виконаного дослідження

До організаційних вимог до реалізації архітектур входять:

- Формулювання чітких вимог до функціональності та взаємодії системи.
- Використання ефективних методів розробки та тестування, таких як ітеративна розробка, контроль версій, автоматизоване тестування.
- Залучення команди розробників з досвідом відповідних технологій та архітектур.
- Забезпечення безпеки даних та конфіденційності клієнтської інформації.
- Планування ресурсів та витрат для реалізації, впровадження та підтримки системи.

Ці вимоги допоможуть забезпечити успішну реалізацію та функціонування обраної архітектури в проектованій експертній системі моніторингу GSM сигналу.

ВИСНОВКИ

Під час виконання кваліфікаційної магістерської роботи, було застосовано як і навички та вміння здобуті під час навчання в інституті, так і знання отримані під час виконання роботи. Обрана тема дозволила зрозуміти принципи методів моніторингу GSM сигналів, та реалізувати власну концепцію аналізу даних.

Було створенно різні програмні рішення які планується подальше вдосконалювати та розвивати. Програмні рішення включили в себе розробку зручних WEB інтерфейсів для користувачів, написання програмних модулів для мікроконтролера STM-32, модулів з реалізацією систем ШІ.

Також було виконано побудове 3D креслень у програмі Solidworks, реалізовано велику кількість апаратних рішень та багато іншого.

Отримані навички під час виконання магістерської роботи дозволяють у майбутньому створювати та розвивати власні проекти, що будуть мати вплив на економічну складову країни.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Опис стандарту GSM [Електронний ресурс] – Режим доступу до ресурсу: <http://glushilka.narod.ru/Jam/GSM.pdf>.
2. Mouly M. The GSM System for Mobile Communications. / М. Mouly, М. В. Pautet., 1992.
3. Шнайер Б. Прикладна криптографія / Брюс Шнайер.
4. Кузнецова А. И. Безпека мереж зв'язку стандарту GSM / А. И. Кузнецова.
5. Стаття 145 Кодексу України про адміністративні правопорушення. [Електронний ресурс] – Режим доступу до ресурсу: <https://urst.com.ua/kupap/st-145>.
6. Кессак [Електронний ресурс] – Режим доступу до ресурсу: <https://he.wikipedia.org/wiki/Grøstl#גריפוד>.
7. Голдовський М. В. Автоматизована система ідентифікації джерел радіовипромінювання. [Електронний ресурс] / М. В. Голдовський – Режим доступу до ресурсу: https://er.nau.edu.ua/bitstream/NAU/53699/1/Кваліфікаційна%20робота%20Голдовський_М..pdf. Набойщикова Є. О., Осипенко В. В. Архітектура інформаційних систем : thesis. 2020. URL: <https://er.knutd.edu.ua/handle/123456789/16600> (дата звернення: 11.06.2023).
8. Детектор БПЛА [Електронний ресурс] – Режим доступу до ресурсу: <https://inlab.kyiv.ua/shop/detektor-droniv>.
9. Ластівчин хвіст (з'єднання) [Електронний ресурс] – Режим доступу до ресурсу: <http://protect.gost.ru/document.aspx?control=7&id=130005>.
10. КАТАЛОГ НАЦІОНАЛЬНИХ СТАНДАРТІВ ТА КОДЕКСІВ УСТАЛЕНОЇ ПРАКТИКИ [Електронний ресурс] – Режим доступу до ресурсу: <http://uas.org.ua/ua/natsionalniy-fond-normativnih-dokumentiv/katalog-normativnih-dokumentiv-2/>
11. Defining JPA Entities [Електронний ресурс] – Режим доступу до ресурсу: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwi_2MyFwo6GAxUOwAIHHZgyCYwQFnoECBQQAQ&url=https%3A%2F%2Fwww.baeldung.com%2Fjpa-entities&usg=AOvVaw3-C-Rs3ltWaU_7BPIy2vEO&opi=89978449.
12. двигатели [Електронний ресурс] – Режим доступу до ресурсу: https://elprivod.nmu.org.ua/files/automaticED/Емельянов_Шаговые%20двигатели.pdf.
13. Помазун О. М. Створення розумного світильника як системи інтернету речей [Електронний ресурс] / О. М. Помазун, Д. С. Білокопитов, Р. К. Синицький

– Режим доступу до ресурсу:

<https://www.calameo.com/read/00316837234120fe2934e>.

14. Набойщикова Є. О., Осипенко В. В. Архітектура інформаційних систем : thesis. 2020. URL: <https://er.knutd.edu.ua/handle/123456789/16600> (дата звернення: 11.06.2023).
15. Бабак О., Ісак Л. АРХИТЕКТУРА КОМП'ЮТЕРНИХ СИСТЕМ ПЕРСПЕКТИВИ РОЗВИТКУ. Grail of Science. 2023. № 26. С. 258–260. URL: <https://doi.org/10.36074/grail-of-science.14.04.2023.046> (дата звернення: 11.06.2023).
16. АРХИТЕКТУРА КЛІЄНТ-СЕРВЕР на основі додатка відділу аспірантури та докторантури ОНАХТ / Д. О. Скалій та ін. Automation of technological and business processes. 2018. Т. 10, № 2. URL: <https://doi.org/10.15673/atbp.v10i2.974> (дата звернення: 11.06.2023).
17. Kelly S. Model View Controller. Python, PyGame and Raspberry Pi Game Development. Berkeley, CA, 2016. P. 141–151. URL: https://doi.org/10.1007/978-1-4842-2517-2_17 (date of access: 11.06.2023).
18. Ekblom R. Applied Representational State Transfer : thesis. 2011. URL: <http://urn.kb.se/resolve?urn=urn:nbn:se:umu:diva-39553> (date of access: 11.06.2023).
19. Sun Y. Single-Page Applications. Practical Application Development with AppRun. Berkeley, CA, 2019. P. 141–162. URL: https://doi.org/10.1007/978-1-4842-4069-4_7 (date of access: 11.06.2023).
20. Shemaiev V., Liashenko I., Stefantsev S. Формалізація відкритої галузі знань розподіленої системи управління. Сучасні інформаційні технології у сфері безпеки та оборони. 2019. Т. 34, № 1. С. 64–68. URL: <https://doi.org/10.33099/2311-7249/2019-34-1-64-68> (дата звернення: 11.06.2023).

ДОДАТКИ

Додаток А Фрагменти програмної реалізації інтерфейсу користувача

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>GSM Frequency Band Selection</title>
  <link rel="stylesheet" href="Selection-style.css">
</head>
<body>
  <div class="container">
    <div class="header">
      <h1>Оберіть необхідні GSM частоти</h1>
    </div>
    <div class="frequency-bands">
      <div class="frequency-band" data-band="GSM-900">
        <h2>GSM-900</h2>
        <p>880 - 915 MHz (Up) | 925 - 960 MHz (Down)</p>
        <input type="radio" name="band" id="gsm-900"
value="GSM-900">
        <label for="gsm-900">Select</label>
      </div>
      <div class="frequency-band" data-band="GSM-1800">
        <h2>GSM-1800</h2>
        <p>1805 - 1880 MHz (Up) | 1885 - 1920 MHz
(Down)</p>
        <input type="radio" name="band" id="gsm-1800"
value="GSM-1800">
        <label for="gsm-1800">Select</label>
      </div>
      <div class="frequency-band" data-band="GSM-1900">
        <h2>GSM-1900</h2>
        <p>1900 - 1920 MHz (Up) | 1925 - 1945 MHz
(Down)</p>
        <input type="radio" name="band" id="gsm-1900"
value="GSM-1900">

```

```

        <label for="gsm-1900">Select</label>
    </div>
    <div class="frequency-band" data-band="E-GSM">
        <h2>E-GSM</h2>
        <p>900 - 915 MHz (Up) | 925 - 960 MHz (Down)</p>
        <input type="radio" name="band" id="e-gsm"
value="E-GSM">
        <label for="e-gsm">Select</label>
    </div>
    <div class="frequency-band" data-band="PCS">
        <h2>PCS</h2>
        <p>1850 - 1910 MHz (Up) | 1930 - 1990 MHz
(Down)</p>
        <input type="radio" name="band" id="pcs"
value="PCS">
        <label for="pcs">Select</label>
    </div>
    </div>
    <button id="select-button">Select Frequency
Band</button>
    </div>
    <script src="script.js"></script>
</body>
</html>

```

Додаток Б Фрагменти програмного коду стилів

```
body {
    width: 100vw;
    height: 100vh;
    margin: 0 auto;
    background: #000;
}

body, div {
    display: flex;
    align-items: center;
    justify-content: center;
}

.radar-scale {
    margin-top: 100px;
    width: 600px;
    height: 600px;
    border-radius: 50%;
    background-image:
        linear-gradient(
            rgba(97,197,203,.15) 1px,
            transparent 1px
        ),
        linear-gradient(
            to right,
            rgba(97,197,203,.15) 1px,
            transparent 1px
        );
    background-size: 10px 10px;
    box-shadow:
        inset 5px 5px 40px 10px rgba(0,0,0,0.5),
        inset -5px 5px 40px 10px rgba(0,0,0,0.5),
        inset -5px -5px 40px 10px rgba(0,0,0,0.5),
        inset 5px -5px 40px 10px rgba(0,0,0,0.5);
}
```

```
}
```

```
.radar-gradient-bg {  
  width: 600px;  
  height: 600px;  
  border-radius: 50%;  
  background-image:  
  radial-gradient(  
    circle at center center,  
    rgba(97, 197, 203, .55) 20%,  
    rgba(97, 197, 203, .2) 50%,  
    rgba(97, 197, 203, .05) 65%,  
    rgba(97, 197, 203, .01) 70%  
  );  
}
```

```
.radar-main-border {  
  width: 540px;  
  height: 540px;  
  border-top: solid 4px rgba(97,197,203,.7);  
  border-bottom: solid 4px rgba(97,197,203,.7);  
  border-left: solid 4px rgba(97,197,203,.2);  
  border-right: solid 4px rgba(97,197,203,.2);  
  border-radius: 50%;  
}
```

```
.radar-scale-circles {  
  width: 470px;  
  height: 470px;  
  border: solid 1px rgb(99, 220, 226);  
  border-radius: 50%;  
  background-image:  
  repeating-radial-gradient(  
    circle at center center,  
    rgba(97,197,203),  
    transparent
```

```
        4px,  
        transparent 31px,  
        transparent 20px  
    );  
    opacity: 0.8;  
}  
  
.radar-scale-lines {  
    position: relative;  
    width: 435px;  
    height: 435px;  
    border: solid 1px rgb(99,220,226);  
    border-radius: 50%;  
}  
  
.radar-scale-lines > span {  
    position: absolute;  
    top: 0;  
    left: 50%;  
    margin-left: -1px;  
    width: 2px;  
    height: 100%;  
    background: rgba(97,197,203,.5);  
    transform: rotate(calc(var(--index) * 30deg));  
}  
  
.radar-scale-lines span span {  
    position: absolute;  
    top: -40px;  
    left: 0;  
    width: 30px;  
    height: 15px;  
    margin-left: -15px;  
    text-align: center;  
    font-family: Arial, Helvetica, sans-serif;  
    font-weight: bold;
```

```
    font-size: 14px;
    color: rgba(97,197,203);
    transform: rotate(calc(var(--index) * -30deg));
}

.radar-scale-lines span span:nth-child(2) {
    top: initial;
    bottom: -40px;
}

.radar {
    position: absolute;
    width: 435px;
    height: 435px;
    border-radius: 50%;
    overflow: hidden;
    animation: radar-rotation 3s infinite linear;
}

.radar::after {
    content: ' ';
    position: absolute;
    top: 40.5%;
    width: 30px;
    height: 100px;
    background: linear-gradient(
        90deg,
        rgb(97,197,203),
        transparent 80%
    );
    transform: perspective(3px) rotateX(19deg);
}

.vehicles {
    position: absolute;
    width: 435px;
    height: 435px;
```

```
        border-radius: 50%;
        overflow: hidden;
    }

    .vehicles div {
        position: absolute;
        width: 150px;
        height: 150px;
    }

    .vehicles div span{
        position: absolute;
        top: 50%;
        left: 50%;
        width: 10px;
        height: 10px;
        border-radius: 50%;
        background: rgba(243,39,39);
    }

    @keyframes radar-rotation {
        0%{
            transform: rotate(0deg);
        }
        100%{
            transform: rotate(360deg);
        }
    }
}
```

Додаток В Фрагменти програмного коду форми авторизації

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Login Page</title>
  <link rel="stylesheet" href="login-style.css">
</head>
<body>
  <div class="container">
    <div class="login-box">
      <h2>Login</h2>
      <form action="#">
        <div class="input-group">
          <label for="username">Username:</label>
          <input type="text" id="username"
name="username" placeholder="Enter your username" required>
        </div>
        <div class="input-group">
          <label for="password">Password:</label>
          <input type="password" id="password"
name="password" placeholder="Enter your password" required>
        </div>
        <button type="submit" class="login-
button">Login</button>
      </form>
      <a href="#" class="forgot-password">Forgot
password?</a>
      <p class="signup-link">Don't have an account? <a
href="#">Sign up</a></p>
    </div>
  </div>
</body>
</html>

```

Додаток Г Фрагменти програмного коду радару

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Radar</title>
  <link rel="stylesheet" href="/css/style.css">
</head>
<body>
  <div class="radar-scale">
    <div class="radar-gradient-bg">
      <div class="radar-main-border">
        <div class="radar-scale-circles">
          <div class="radar-scale-lines">
            <span style="--index: 0;">
              <span>0</span>
              <span>180</span>
            </span>
            <span style="--index: 1;">
              <span>30</span>
              <span>210</span>
            </span>
            <span style="--index: 2;">
              <span>60</span>
              <span>240</span>
            </span>
            <span style="--index: 3;">
              <span>90</span>
              <span>270</span>
            </span>
          </div>
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```


Додаток Д Фрагменти програмного коду управління антеною GSM

```
#!/usr/bin/env python3
import serial
import numpy as np
import pylab as pl
import struct
from serial.tools import list_ports

VID = 0x0483 #1155
PID = 0x5740 #22336

# Get tinysa device automatically
def getport() -> str:
    device_list = list_ports.comports()
    for device in device_list:
        if device.vid == VID and device.pid == PID:
            return device.device
    raise OSError("device not found")

REF_LEVEL = (1<<9)

class tinySA:
    def __init__(self, dev = None):
        self.dev = dev or getport()
        self.serial = None
        self._frequencies = None
        self.points = 101

    @property
    def frequencies(self):
        return self._frequencies

    def set_frequencies(self, start = 1e6, stop = 350e6, points
= None):
        if points:
            self.points = points
```

```

self.points) self._frequencies = np.linspace(start, stop,

def open(self):
    if self.serial is None:
        self.serial = serial.Serial(self.dev)

def close(self):
    if self.serial:
        self.serial.close()
    self.serial = None

def send_command(self, cmd):
    self.open()
    self.serial.write(cmd.encode())
    self.serial.readline() # discard empty line
# print(self.serial.readline()) # discard empty line

def cmd(self, text):
    self.open()
    self.serial.write((text + "\r").encode())
    self.serial.readline() # discard empty line
    data = self.fetch_data()
    return data
# self.serial.readline() # discard empty line

def set_sweep(self, start, stop):
    if start is not None:
        self.send_command("sweep start %d\r" % start)
    if stop is not None:
        self.send_command("sweep stop %d\r" % stop)

def set_span(self, span):
    if span is not None:
        self.send_command("sweep span %d\r" % span)

def set_center(self, center):

```

```
        if center is not None:
            self.send_command("sweep center %d\r" % center)

def set_level(self, level):
    if level is not None:
        self.send_command("level %d\r" % level)

def set_output(self, on):
    if on is not None:
        if on:
            self.send_command("output on\r")
        else:
            self.send_command("output off\r")

def set_low_output(self):
    self.send_command("mode low output\r")

def set_low_input(self):
    self.send_command("mode low input\r")

def set_high_input(self):
    self.send_command("mode high input\r")

def set_frequency(self, freq):
    if freq is not None:
        self.send_command("freq %d\r" % freq)

def measure(self, freq):
    if freq is not None:
        self.send_command("hop %d 2\r" % freq)
        data = self.fetch_data()
        for line in data.split('\n'):
            if line:
                return float(line)

def temperature(self):
    self.send_command("k\r")
```

```

    data = self.fetch_data()
    for line in data.split('\n'):
        if line:
            return float(line)

def rbw(self, data=0):
    if data == 0:
        self.send_command("rbw auto\r")
        return
    if data < 1:
        self.send_command("rbw %f\r" % data)
        return
    if data >= 1:
        self.send_command("rbw %d\r" % data)

def fetch_data(self):
    result = ''
    line = ''
    while True:
        c = self.serial.read().decode('utf-8')
        if c == chr(13):
            next # ignore CR
        line += c
        if c == chr(10):
            result += line
            line = ''
            next
        if line.endswith('ch>'):
            # stop on prompt
            break
    return result

# def fetch_array(self, sel):
#     self.send_command("data %d\r" % sel)
#     data = self.fetch_data()
#     x = []
#     for line in data.split('\n'):

```

```

#             if line:
#                 x.extend([float(d)         for         d         in
line.strip().split(' ')])
#                 return np.array(x[0::2]) + np.array(x[1::2]) * 1j

#     def fetch_gamma(self, freq = None):
#         if freq:
#             self.set_frequency(freq)
#             self.send_command("gamma\r")
#             data = self.serial.readline()
#             d = data.strip().split(' ')
#             return (int(d[0])+int(d[1])*1.j)/REF_LEVEL

def resume(self):
    self.send_command("resume\r")

def pause(self):
    self.send_command("pause\r")

def marker_value(self, nr = 1):
    self.send_command("marker %d\r" % nr)
    data = self.fetch_data()
    line = data.split('\n')[0]
#     print(line)
    if line:
        dl = line.strip().split(' ')
        if len(dl) >= 4:
            d = line.strip().split(' ')[3]
            return float(d)
    return 0

def data(self, array = 2):
    self.send_command("data %d\r" % array)
    data = self.fetch_data()
    x = []
    for line in data.split('\n'):
        if line:

```

```

        d = line.strip().split(' ')
        x.append(float(line))
    return np.array(x)

def fetch_frequencies(self):
    self.send_command("frequencies\r")
    data = self.fetch_data()
    x = []
    for line in data.split('\n'):
        if line:
            x.append(float(line))
    self._frequencies = np.array(x)

def send_scan(self, start = 1e6, stop = 900e6, points =
None):
    if points:
        self.send_command("scan %d %d %d\r"%(start,
stop, points))
    else:
        self.send_command("scan %d %d\r"%(start, stop))

def scan(self):
    segment_length = 101
    array0 = []
    array1 = []
    if self._frequencies is None:
        self.fetch_frequencies()
    freqs = self._frequencies
    while len(freqs) > 0:
        seg_start = freqs[0]
        seg_stop = freqs[segment_length-1] if len(freqs)
>= segment_length else freqs[-1]
        length = segment_length if len(freqs) >=
segment_length else len(freqs)
        #print((seg_start, seg_stop, length))
        self.send_scan(seg_start, seg_stop, length)
        array0.extend(self.data(0))
        array1.extend(self.data(1))

```



```

        help="scan          by          script",
metavar="SCAN")
        parser.add_option("-S", "--start", dest="start",
                           type="float", default=1e6,
                           help="start          frequency",
metavar="START")
        parser.add_option("-E", "--stop", dest="stop",
                           type="float", default=900e6,
                           help="stop          frequency",
metavar="STOP")
        parser.add_option("-N", "--points", dest="points",
                           type="int", default=101,
                           help="scan          points",
metavar="POINTS")
        parser.add_option("-P", "--port", type="int",
dest="port",
                           help="port", metavar="PORT")
        parser.add_option("-d", "--dev", dest="device",
                           help="device node", metavar="DEV")
        parser.add_option("-v", "--verbose",
                           action="store_true",
dest="verbose", default=False,
                           help="verbose output")
        parser.add_option("-C", "--capture", dest="capture",
                           help="capture current display to
FILE", metavar="FILE")
        parser.add_option("-e", dest="command", action="append",
                           help="send          raw          command",
metavar="COMMAND")
        parser.add_option("-o", dest="save",
                           help="write          CSV          file",
metavar="SAVE")
        (opt, args) = parser.parse_args()

        nv = tinySA(opt.device or getport())

        if opt.command:
            print(opt.command)
            for c in opt.command:
                nv.send_command(c + "\r")

```

```
        data = nv.fetch_data()
        print(data)

    if opt.capture:
        print("capturing...")
        img = nv.capture()
        img.save(opt.capture)
        exit(0)

#     nv.set_port(opt.port)
    if opt.start or opt.stop or opt.points:
        nv.set_frequencies(opt.start, opt.stop, opt.points)
#     plot = opt.plot
    if opt.plot or opt.save or opt.scan:
        p = int(opt.port) if opt.port else 0
        if opt.scan or opt.points > 101:
            s = nv.scan()
            s = s[p]
        else:
            if opt.start or opt.stop:
                nv.set_sweep(opt.start, opt.stop)
            nv.fetch_frequencies()
            s = nv.data(p)
#         nv.fetch_frequencies()
    if opt.save:
        nv.writeCSV(s, opt.save)
    if opt.plot:
        nv.logmag(s)
        pl.show()
```

Додаток Е Фрагменти програмного коду для системи III

```
package com.example.demo;

import org.deeplearning4j.nn.conf.MultiLayerConfiguration;
import org.deeplearning4j.nn.conf.NeuralNetConfiguration;
import org.deeplearning4j.nn.conf.layers.DenseLayer;
import org.deeplearning4j.nn.conf.layers.OutputLayer;
import org.deeplearning4j.nn.multilayer.MultiLayerNetwork;
import org.deeplearning4j.nn.weights.WeightInit;
import org.nd4j.linalg.activations.Activation;
import org.nd4j.linalg.api.ndarray.INDArray;
import org.nd4j.linalg.factory.Nd4j;
import org.nd4j.linalg.learning.config.Sgd;
import org.nd4j.linalg.lossfunctions.LossFunctions;
import org.springframework.stereotype.Service;

/**
 * Service for predicting movements based on latitude and
longitude inputs.
 * @author Dima
 */
@Service
public class MovementPredictionService {
    private MultiLayerNetwork model;

    public MovementPredictionService() {
        int numInput = 2;
        int numOutputs = 2;
```

```

int numHiddenNodes = 20;

double learningRate = 0.01;

        MultiLayerConfiguration      conf      =      new
NeuralNetConfiguration.Builder()

        .updater(new Sgd(learningRate))

        .weightInit(WeightInit.XAVIER)

        .list()

        .layer(0,                                new
DenseLayer.Builder().nIn(numInput).nOut(numHiddenNodes)

                .activation(Activation.RELU)

                .build())

        .layer(1,                                new
OutputLayer.Builder(LossFunctions.LossFunction.MSE)

                .activation(Activation.IDENTITY)

.nIn(numHiddenNodes).nOut(numOutputs).build())

        .build();

        this.model = new MultiLayerNetwork(conf);
        this.model.init();
    }

    public void train(double[][] inputData, double[][]
outputData) {

        INDArray input = Nd4j.create(inputData);
        INDArray output = Nd4j.create(outputData);
        for (int i = 0; i < 1000; i++) {

            model.fit(input, output);

```

```
        }  
    }  
  
    public double[] predictNextMovement(double[]  
lastTwoMovements) {  
        INDArray input = Nd4j.create(lastTwoMovements);  
        INDArray predicted = model.output(input);  
        return predicted.toDoubleVector();  
    }  
}
```



Имя пользователя: Інформаційних систем в економіці Шкуратовська Те...	ID проверки: 1016271765
Дата проверки: 22.05.2024 01:30:24 EEST	Тип проверки: Doc vs Internet + Library
Дата отчета: 22.05.2024 07:00:31 EEST	ID пользователя: 100005745

Название файла: Білокопитов Д.С. ІШІ-601
 Количество страниц: 57 Количество слов: 8330 Количество символов: 65555 Размер файла: 2.18 MB ID файла: 1016061449

Обнаружены модификации текста (могут влиять на процент совпадений)

9% Совпадения

Наибольшее совпадение: 2.03% с источником из Библиотеки (ID файла: 1014813427)

3.45% Источники из Интернета	212	Страница 59
8.26% Источники из Библиотеки	415	Страница 60

0% Цитат

- Исключение цитат выключено
- Исключение списка библиографических ссылок выключено

0% Исключений

Нет исключенных источников

Модификации

Обнаружены модификации текста. Подробная информация доступна в онлайн-отчете.

Замененные символы	9
Подозрительное форматирование	9 страниц