Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

# МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Захищена система оповіщення населення про ракетну небезпеку в автоматичному та «ручному» режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних»

Виконав: ст. 2-го курсу, групи КІТС-22 мз
спеціальності 125– Кібербезпека
Освітня програма – Кібербезпека
інформаційних технологій та систем
(шифр і назва напряму підготовки, спеціальності)

Пономарьов В.О.
(прізвище та ініціали)

Керівник: д.т.н., проф. каф. МБІС
Яремчук Ю.Є.
(прізвище та ініціали)
«____» _____ 2024 р.

Опонент: к.т.н., доцент, каф. ОТ
Колесник І.С._____
(прізвище та ініціали)
«____» _____ 2024 р.

**Допущено до захисту**
Голова секції УБ кафедри МБІС

Юрій ЯРЕМЧУК
“_____” _____2024 р.

Вінниця ВНТУ – 2024 рік

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

Рівень вищої освіти ІІ-й (магістерський)
Галузь знань 12 – Інформаційні технології
Спеціальність 125 – Кібербезпека
Освітньо-професійна програма - Кібербезпека інформаційних технологій та систем

**ЗАТВЕРДЖУЮ**
**Голова секції УБ, кафедра МБІС**
_____ **Юрій ЯРЕМЧУК**
“_____” _____2024 р.

**З А В Д А Н Н Я**
**на магістерську кваліфікаційну роботу студенту**
Пономарьов Владислав Олександрович
(прізвище, ім`я, по-батькові)

1. Тема роботи:

«Захищена система оповіщення населення про ракетну небезпеку в автоматичному та «ручному» режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних»
Керівник роботи_: Яремчук Ю.Є. д.т.н., професор каф. МБІС
(прізвище, ім`я, по-батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від “ 11” березня 2024 року № 81
2. Строк подання студентом роботи_за тиждень до захисту.__
3. Вихідні дані до роботи:
Стандарти, електронні джерела, підручники та наукові статті по темі, які стосуються теми магістерської кваліфікаційної роботи.
4. Зміст текстової частини:
У першому розділі розглянуто теоретичні основи захисту системи оповіщення населення про ракетну небезпеку. Висвітлено важливість використання різних видів систем оповіщення, критичні елементи захисту інформації, значення резервних каналів зв’язку та методів шифрування даних.
У другому розділі детально описано процес створення захищеної системи оповіщення. Розглянуто основи проектування системи, спроектовано схему роботи пристрою. У третьому розділі здійснено програмну реалізацію системи. Обґрунтовано вибір мови програмування та середовища розробки, представлено програмну реалізацію серверної та апаратної частин, а також проведено аналіз і тестування реалізованих засобів.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов`язкових креслень)

У другому розділі магістерської кваліфікаційної роботи наведено 3 малюнка, у третьому розділі – 10 малюнків.

6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| Основна частина | | | |
| I | Яремчук Ю.Є. д.т.н., професор каф. МБІС | | |
| II | Яремчук Ю.Є. д.т.н., професор каф. МБІС | | |
| III | Яремчук Ю.Є. д.т.н., професор каф. МБІС | | |
| Економічна частина | | | |
| IV | Причепа І.В., к.е.н., доц. каф. ЕПВМ | | |

7.Дата видачі завдання _____2024 р.

## КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів магістерської кваліфікаційної роботи | Строк виконання етапів роботи | | Примітка |
|---|---|---|---|---|
| 1. | Визначення напрямку магістерської роботи, формулювання теми | 1.03.2024 | 15.03.2024 | |
| 2. | Аналіз предметної області обраної теми | 15.03.2024 | 1.04.2024 | |
| 3. | Розробка роботи | 1.04.2024 | 10.04.2024 | |
| 4. | Написання магістерської роботи на основі розробленої теми | 10.04.2024 | 20.04.2024 | |
| 5. | Передзахист магістерської кваліфікаційної роботи | 20.04.2024 | 10.05.2024 | |
| 6. | Виправлення, уточнення, корегування магістерської кваліфікаційної роботи | 10.05.2024 | 4.06.2024 | |
| 7. | Захист магістерської кваліфікаційної роботи | 10.06.2024 | 10.06.2024 | |

**Студент** _____ Пономарьов В.О.
(підпис)

**Керівник роботи** _____ Яремчук Ю.Є.
(підпис)

# АНОТАЦІЯ

УДК 004.56.5(043.2)

Пономарьов В.О.. Захищена система оповіщення населення про ракетну небезпеку в автоматичному та “ручному” режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних.

Магістерська кваліфікаційна робота зі спеціальності 125 – «Кібербезпека», освітня програма «Кібербезпека інформаційних технологій та систем». Вінниця: ВНТУ, 2024. 116с.

На укр.мові. Бібліогр.: 41 назв; рис.: 26; табл. 17

У магістерській кваліфікаційній роботі розглянута розробка захищеної системи оповіщення населення про ракетну небезпеку, яка функціонує в автоматичному та «ручному» режимах.

У першому розділі розглянуто теоретичні основи захисту інформації в системах оповіщення, обґрунтовано важливість використання резервних каналів зв'язку та методів шифрування даних, а також проаналізовано різні види зв’язків для обміну інформацією. Завершується розділ постановкою задачі дослідження.

У другому розділі детально описано процес створення захищеної системи оповіщення. Розглянуто основи проектування системи, спроектовано схему роботи пристрою та розроблено алгоритм його функціонування. Особливу увагу приділено проектуванню системи резервних каналів обміну.

Третій розділ присвячений програмній реалізації системи. Обґрунтовано вибір мови програмування та середовища розробки, представлено програмну реалізацію серверної та апаратної частин.

У четвертому розділі здійснено економічний аналіз розробки. Оцінено комерційний потенціал програмного забезпечення, прогнозовано витрати на виконання наукової роботи та впровадження її результатів.

Ключові слова: оповіщення населення про ракетну небезпеку, односторонній зв’язок, резервні канали, шифрування.

## ABSTRACT

V.O. Ponomarov. A protected system for notifying the population about missile danger in automatic and "manual" modes using one-way communication, backup exchange channels and data encryption.

Master's qualification thesis on specialty 125 - "Cybersecurity", educational program "Cybersecurity of information technologies and systems". Vinnytsia: VNTU, 2024. 130p.

In the Ukrainian language. Bibliography: 41 titles; Fig.: 26; table 17

The master's thesis deals with the development of a secure system for notifying the population about missile danger, which functions in automatic and "manual" modes.

In the first chapter, the theoretical foundations of information protection in notification systems are considered, the importance of using backup communication channels and data encryption methods is substantiated, and various types of communications for information exchange are analyzed. The chapter ends with a statement of the research problem.

The second section describes in detail the process of creating a secure notification system. The fundamentals of system design were considered, the scheme of the device was designed, and the algorithm of its functioning was developed. Special attention is paid to the design of the system of backup exchange channels.

The third section is devoted to the software implementation of the system. The choice of programming language and development environment is justified, the software implementation of the server and hardware parts is presented, and the analysis and testing of the implemented tools is performed.

In the fourth chapter, an economic analysis of the development is carried out. The commercial potential of the software was assessed, the costs of carrying out scientific work and the implementation of its results were predicted.



Keywords: alerting the population about missile, one-way communication, backup channels, encryption.

**ЗМІСТ**

# ВСТУП

**Актуальність.** У сучасному світі зростання загрози ракетних ударів, особливо у регіонах з високою ймовірністю військових конфліктів, викликає нагальну потребу в надійних системах оповіщення населення. За останні роки, значне зростання інтенсивності ракетних атак та застосування сучасних технологій зброї масового ураження призвели до значних жертв серед населення та руйнувань інфраструктури. Тому забезпечення своєчасного оповіщення є критично важливим для зменшення людських і матеріальних втрат.

Застарілі системи оповіщення часто не здатні забезпечити належну швидкість і точність передачі інформації про загрозу, що може призвести до катастрофічних наслідків. Водночас, сучасні технології надають можливості для створення більш надійних і ефективних систем, що використовують односторонній зв'язок, резервні канали обміну та шифрування даних.

Захищені системи оповіщення, які здатні працювати в автоматичному та «ручному» режимах, є важливими для забезпечення безпеки як в мирний час, так і в умовах війни. Використання одностороннього зв'язку унеможливлює перехоплення і модифікацію повідомлень з боку противника, що є особливо важливим у військових умовах.

Резервні канали обміну інформацією забезпечують додаткову надійність системи, дозволяючи уникнути втрат інформації у випадку виходу з ладу основних каналів зв'язку. Методи шифрування даних, у свою чергу, гарантують конфіденційність і цілісність переданої інформації, що є критичним у умовах підвищеної загрози кібернападів.

Таким чином, створення захищеної системи оповіщення населення про ракетну небезпеку, дозволить значно підвищити рівень безпеки населення, забезпечити своєчасне реагування на загрози і мінімізувати наслідки ракетних атак.

**Мета і задачі дослідження.** Метою даної роботи є розробка захищеної системи оповіщення населення про ракетну небезпеку, що функціонує в

автоматичному та «ручному» режимах, з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних.

**Задачами дослідження є:**

– аналіз теоретичних основ захисту інформації в системах оповіщення.

– визначення важливості та ролі резервних каналів зв’язку та методів шифрування даних.

– проектування системи резервних каналів обміну інформацією.

– розробка схеми та алгоритму роботи пристрою оповіщення населення про ракетну небезпеку.

– програмна реалізація серверної та апаратної частин системи.

– аналіз та тестування розроблених засобів.

– економічне обґрунтування розробки програмного забезпечення та прогнозування комерційних ефектів.

**Об’єктом дослідження** є системи оповіщення населення про надзвичайні ситуації.

**Предметом дослідження** є методи і засоби забезпечення захищеності інформації в системах оповіщення населення про ракетну небезпеку, зокрема використання одностороннього зв'язку, резервних каналів обміну та шифрування даних.

**Новизна роботи** полягає в розробці інтегрованої системи оповіщення, що поєднує автоматичний та «ручний» режими оповіщення з використанням сучасних методів шифрування та резервних каналів зв’язку. Це дозволяє підвищити надійність і безпеку передачі інформації в умовах загрози ракетного удару.

**Практична цінність** роботи полягає у можливості впровадження розробленої системи в реальні умови для підвищення рівня безпеки населення. Запропоновані рішення можуть бути використані державними та приватними установами для створення ефективних систем оповіщення, що мінімізують ризики та наслідки ракетних атак.

# 1 ТЕОРЕТИЧНІ ОСНОВИ ЗАХИЩЕННЯ СИСТЕМИ ОПОВІЩЕННЯ НАСЕЛЕННЯ ПРО РАКЕТНУ НЕБЕЗПЕКУ

У першому розділі роботи розглядатимуться теоретичні аспекти захисту системи оповіщення населення про ракетну небезпеку. Аналізуватимуться важливість використання систем оповіщення, критичні елементи захисту інформації, роль резервних каналів зв’язку, методи шифрування даних та вибір зв’язку для обміну інформацією.

Висновки розділу підкреслять основні висновки та постановку завдань для подальшого дослідження.

## 1.1 Важливість використання та види систем оповіщення населення

Системи оповіщення населення мають вирішальне значення для забезпечення безпеки та захисту громадян у випадку надзвичайних ситуацій, таких як ракетна небезпека. Ці системи надають можливість швидкого та ефективного сповіщення громадян про потенційні загрози та надають необхідні інструкції для забезпечення їх власної безпеки.

Важливість використання систем оповіщення населення полягає в забезпеченні безпеки та захисту громадян у випадку надзвичайних ситуацій, таких як ракетна небезпека. Ці системи надають можливість швидкого та ефективного сповіщення населення про потенційні загрози та надають інструкції щодо дій для захисту власної безпеки [1].

Існують різноманітні види систем оповіщення населення, включаючи:

– системи автоматизованого масового повідомлення (САМП);

Системи автоматизованого масового повідомлення (САМП) є ключовим інструментом в системах оповіщення населення через їх здатність ефективно та оперативно інформувати громадян про надзвичайні події. Ці системи використовують різноманітні канали зв'язку, такі як мобільні додатки, SMS, електронна пошта, соціальні мережі та автоматизовані голосові дзвінки, для масового розсилання повідомлень населенню.

Мобільні додатки надають зручний та швидкий спосіб отримання повідомлень про надзвичайні ситуації безпосередньо на смартфони користувачів. Вони можуть містити інформацію про поточну ситуацію, інструкції щодо безпеки та поради щодо дій у разі небезпеки.

SMS-повідомлення можуть бути ефективним каналом для швидкого розсилання коротких інформаційних повідомлень громадянам, навіть якщо вони не мають доступу до Інтернету або не користуються мобільними додатками.

Електронна пошта також може використовуватися для розсилання повідомлень про надзвичайні ситуації, забезпечуючи більш деталізовану інформацію та можливість надсилання додаткових матеріалів або посилань на офіційні джерела інформації.

Соціальні мережі стали популярним каналом для отримання новин та інформації, тому розсилка повідомлень через ці платформи може забезпечити широкий охоплення аудиторії та швидку реакцію на надзвичайні ситуації.

Автоматизовані голосові дзвінки можуть бути ефективним способом інформування людей, особливо тих, хто не має доступу до Інтернету або не вміє користуватися мобільними додатками чи соціальними мережами. Голосові повідомлення можуть містити важливі інструкції та поради щодо дій у небезпеці.

– системи публічного оповіщення;

Системи публічного оповіщення - це важлива складова частина систем оповіщення населення, призначена для ефективного та швидкого інформування громадян у разі надзвичайних ситуацій. Вони включають в себе розміщення сирен та гучномовців у публічних місцях, які можуть відтворювати аварійні сигнали або інструкції для населення.

Сирени, які встановлюються на вулицях, у парках, біля шкіл та інших громадських місць, є дуже ефективним засобом оповіщення в разі надзвичайних ситуацій. Вони можуть відтворювати різні звукові сигнали, що вказують на різні типи небезпеки, такі як пожежа, надзвичайна ситуація чи загроза терористичного акту.

Гучномовці також використовуються для розповсюдження важливої інформації та інструкцій для населення. Вони можуть бути розташовані у громадських приміщеннях, на транспортних вузлах або інших місцях з великою кількістю людей. Гучномовці дозволяють передавати не лише звукові повідомлення, але і інші форми комунікації, такі як голосові інструкції чи інформаційні оголошення.

Ці системи публічного оповіщення є надзвичайно важливими для забезпечення безпеки та захисту населення в разі надзвичайних ситуацій, оскільки вони забезпечують можливість швидкого та ефективного реагування на небезпеку, що може зберегти життя та майно громадян.

– системи електронних табло та попередження на дорогах;

Системи електронних табло та попередження на дорогах є важливим елементом систем оповіщення населення в разі надзвичайних ситуацій. Ці системи встановлюються в населених пунктах та на транспортних маршрутах і використовуються для показу важливої інформації та інструкцій щодо евакуації або захисту [2].

Електронні табло розташовані у стратегічних місцях, таких як громадські площі, вулиці, станції метро, аеропорти тощо. Вони можуть відображати різні види повідомлень, включаючи інформацію про надзвичайні ситуації, інструкції щодо евакуації, попередження про небезпеку та інше. Електронні табло можуть бути програмовані для автоматичного відтворення попередньо підготовлених повідомлень або для відображення актуальної інформації згідно з поточною ситуацією.

Системи попередження на дорогах використовуються для інформування водіїв про надзвичайні ситуації на дорогах. Ці системи можуть бути розташовані біля дорожніх знаків, світлофорів або на спеціальних конструкціях поруч з дорогами. Вони можуть показувати різні типи повідомлень, включаючи попередження про аварії, ремонтні роботи, погодні умови та іншу важливу інформацію для безпечної їзди.

Ці системи електронних табло та попередження на дорогах є ефективними засобами забезпечення безпеки населення в разі надзвичайних ситуацій, оскільки вони дозволяють оперативно та чітко інформувати людей про потенційні загрози та надавати відповідні інструкції для їх захисту [3].

– системи радіо- та телевізійного оповіщення.

Системи радіо- та телевізійного оповіщення є важливим засобом масового інформування та оповіщення населення в разі надзвичайних ситуацій. Мовлення радіо та телевізійних каналів може бути використане для ретрансляції надзвичайних повідомлень та інструкцій для населення.

Радіо та телевізійні канали мають широкий охоплення та можуть доставляти інформацію до великої аудиторії одночасно. Вони є доступними практично для всіх груп населення, незалежно від їх місця перебування чи наявності доступу до інтернету. Такі канали можуть передавати інформацію у реальному часі, що дозволяє оперативно інформувати населення про потенційні загрози або надавати інструкції щодо дій в надзвичайних ситуаціях.

Крім того, радіо та телевізійні канали можуть бути використані для ретрансляції важливих повідомлень на різних мовах чи для різних аудиторій, що дозволяє забезпечити доступність інформації для різних соціокультурних груп населення.

Такі системи є важливим елементом інфраструктури оповіщення населення та дозволяють оперативно та ефективно інформувати громадян про надзвичайні ситуації та надавати рекомендації щодо безпечної поведінки.

Використання систем оповіщення населення має велике значення для забезпечення безпеки та ефективного реагування на надзвичайні ситуації. Різноманітні види систем, такі як системи автоматизованого масового повідомлення, публічного оповіщення, електронних табло та попередження на дорогах, радіо- та телевізійного оповіщення, забезпечують широкий охоплення та доступність інформації для населення у різних ситуаціях. Ці системи дозволяють оперативно та ефективно повідомляти громадян про потенційні загрози та надавати

інструкції щодо дій у надзвичайних обставинах, сприяючи підвищенню рівня безпеки та зменшенню ризиків для населення.

### 1.2 Критичні елементи захисту інформації

У сучасному цифровому середовищі забезпека інформації є критично важливою для успішної діяльності організацій. Ключове значення в цьому процесі мають критичні елементи захисту інформації, які визначають рівень безпеки системи в цілому. Розглянувши їх детально, можна забезпечити ефективний захист від різних загроз інформаційної безпеки.

Критичні елементи захисту інформації - це основні складові, які визначають рівень безпеки та надійності системи в цілому. Розглянемо детально кілька таких елементів:

– ідентифікація і аутентифікація;

Ідентифікація та аутентифікація є важливими елементами захисту інформації, оскільки вони визначають, хто має доступ до системи.

Ідентифікація визначає особистість користувача, надаючи унікальний ідентифікатор або ім'я, за яким його можна впізнати в системі. Це може бути, наприклад, логін, електронна пошта або інший унікальний ідентифікатор.

Аутентифікація встановлює, чи є надана особа тією, за кого себе видає. Зазвичай це виконується за допомогою пароля, ключа, біометричних даних або комбінації цих методів. Ефективна аутентифікація дозволяє перевірити, чи має користувач право на доступ до системи [4].

Застосування надійних методів ідентифікації та аутентифікації є критично важливим для запобігання несанкціонованому доступу до інформації та збереження цілісності даних. Для цього можуть використовуватися різні технології та методи, які забезпечують високий рівень захисту інформації в системі.

– контроль доступу;

Контроль доступу - це ключовий елемент в системах безпеки, який регулює доступ користувачів до різних ресурсів і функцій. Ефективні механізми контролю

доступу призначені для обмеження прав доступу користувачів на основі їхніх ролей, обов'язків та потреб в системі. Ось детальний огляд цього елемента:

– ролі користувачів: Контроль доступу визначається на основі ролей, які відповідають рівню доступу користувача. Користувачам призначаються ролі, які визначають їхні права доступу до різних ресурсів. Наприклад, адміністратор має повний доступ до всіх функцій системи, тоді як звичайний користувач може мати обмежений доступ;

– права доступу: Кожна роль або користувач має набір прав доступу, які визначають, які операції вони можуть виконувати в системі. Ці права включають доступ до конкретних ресурсів, виконання певних дій або взаємодію з певними функціями;

– методи автентифікації: Контроль доступу також включає в себе методи автентифікації, які підтверджують ідентичність користувачів перед наданням доступу. Це може бути здійснено за допомогою паролів, біометричних даних, ключів або інших методів;

– політики безпеки: Ефективна система контролю доступу базується на чітко визначених політиках безпеки, які встановлюють правила доступу до ресурсів і функцій системи. Ці політики визначаються на основі вимог безпеки та внутрішніх правил організації;

– моніторинг і аудит: Контроль доступу також включає в себе моніторинг та аудит доступу, які відслідковують та реєструють дії користувачів в системі. Це дозволяє виявляти відхилення від політик безпеки та реагувати на потенційні загрози.

Ефективний контроль доступу гарантує, що тільки авторизовані користувачі мають доступ до необхідних ресурсів і функцій, зменшуючи ризик несанкціонованого доступу та витоку конфіденційної інформації.

– шифрування даних;

Шифрування даних є одним з фундаментальних елементів безпеки інформації. Цей процес полягає в перетворенні звичайного тексту або даних в

зашифрований вигляд за допомогою математичних алгоритмів, які надають конфіденційність та захист під час передачі або зберігання. Ось детальний огляд процесу шифрування даних:

– конфіденційність і захист даних: Основна мета шифрування полягає в тому, щоб зробити дані нерозбірливими для незаконного доступу. Шифрування дозволяє зберігати та передавати дані безпечно, навіть якщо вони потраплять у руки несанкціонованих осіб, оскільки без ключа шифрування відновлення оригінального вмісту є практично неможливим;

– ключі шифрування: Для шифрування та розшифрування даних використовуються ключі, які є спеціальними послідовностями бітів або символів. Ключі використовуються алгоритмом шифрування для перетворення даних у зашифровану форму, а потім для відновлення оригінального вмісту;

– симетричне та асиметричне шифрування: Існують два основних типи шифрування - симетричне та асиметричне. У симетричному шифруванні один ключ використовується як для шифрування, так і для розшифрування даних, тоді як у асиметричному шифруванні використовуються два ключі: публічний та приватний;

– застосування шифрування: Шифрування використовується в різних сферах інформаційної безпеки, таких як зберігання паролів, захищене з'єднання з веб-сайтами (SSL / TLS), електронна пошта, мережеві протоколи, файли на зберіганні та багато іншого;

– методи шифрування: Існує безліч алгоритмів шифрування, таких як AES, RSA, DES, і багато інших. Вибір конкретного методу шифрування залежить від конкретних потреб та характеристик системи, таких як рівень безпеки та швидкодія.

Шифрування даних є важливим і необхідним елементом в сучасних системах інформаційної безпеки, що дозволяє забезпечити конфіденційність та захист важливої інформації від несанкціонованого доступу.

– моніторинг та аудит;

Моніторинг та аудит є важливими складовими систем безпеки, оскільки вони дозволяють відстежувати та реагувати на незвичайну або підозрілу активність у системі. Розглянемо їх детальніше:

Моніторинг:

– відстеження активності користувачів: Моніторинг дозволяє відстежувати активність користувачів у реальному часі, включаючи їхні дії та взаємодію з системою;

– виявлення аномалій: Шляхом аналізу патернів та поведінки користувачів моніторинг може виявити аномальні або підозрілі дії, які можуть свідчити про потенційні загрози безпеці;

– реагування на інциденти: Правильно налаштований моніторинг дозволяє оперативно реагувати на інциденти безпеки та вживати відповідних заходів для їх усунення.

Аудит:

– запис активності: Аудит забезпечує запис інформації про всі дії, які відбуваються у системі, включаючи авторизації, зміни конфігурацій, доступ до файлів тощо;

– виявлення вразливостей і підозрілих дій: Шляхом аналізу аудит-логів можна виявити вразливості системи та підозрілу активність, яка може свідчити про атаки або несанкціонований доступ;

– забезпечення відповідності: Аудит допомагає впевнитися, що система відповідає вимогам безпеки та стандартам, а також дотримується політик безпеки та регуляторних вимог.

В цілому, моніторинг та аудит є ключовими елементами в системах безпеки, які допомагають вчасно виявляти та реагувати на потенційні загрози, забезпечуючи високий рівень безпеки та відповідність стандартам і вимогам безпеки [5].

– фізична безпека;

Фізична безпека включає широкий спектр заходів для захисту фізичного обладнання, приміщень і інфраструктури від різних загроз. Розглянемо детальніше цей аспект безпеки:

Захист фізичного доступу:

– контроль доступу: Включає установку систем контролю доступу, таких як кард-рідери, біометричні сканери, або використання фізичних ключів для обмеження доступу до приміщень та серверних кімнат.

– моніторинг інтегрованих систем безпеки: Встановлення камер спостереження та сенсорів руху для виявлення та відстеження несанкціонованої активності.

Захист інфраструктури:

– фізичне розташування серверних центрів: Важливо розташовувати серверні приміщення в безпечних місцях, обмежуючи доступ та знижуючи ризик негативного впливу природних катастроф або інших небезпек.

– запобігання крадіжок та знищень: Застосування фізичних заходів безпеки, таких як замки, алюмінієві двері, металеві решітки, щоб запобігти крадіжкам та знищенню обладнання.

Запобігання природним катастрофам:

– будівництво відповідно до стандартів безпеки: Важливо будувати приміщення та інфраструктуру, враховуючи потенційні природні небезпеки, такі як землетруси, повені, пожежі та інші.

– захист від пожеж: Встановлення систем пожежної сигналізації, автоматичних систем гасіння пожежі та розташування пожежних виходів.

Фізична безпека є важливим елементом загальної стратегії безпеки, оскільки вона забезпечує захист від різних загроз, які можуть виникнути через доступ до фізичних об'єктів та інфраструктури. Дотримання відповідних стандартів і реалізація відповідних заходів безпеки дозволяє знизити ризики та забезпечити надійну захист від потенційних загроз.

– заходи безпеки мережі;

Заходи безпеки мережі є критичними для забезпечення цілісності, конфіденційності та доступності мережевих ресурсів. Розглянемо детальніше такі заходи:

– файрволи та інтрузійна детекція (firewalls and intrusion detection):

Файрволи: Це програмне та/або апаратне забезпечення, яке фільтрує трафік мережі на основі заздалегідь визначених правил. Вони блокують небажаний трафік, захищаючи мережу від зовнішніх загроз.

Інтрузійна детекція: Ці системи аналізують мережевий трафік та шукають ознаки атак або несанкціонованої активності в мережі, сповіщаючи адміністраторів про потенційні загрози [6].

– шифрування мережевого трафіку:

Використання протоколів шифрування, таких як SSL/TLS для захисту конфіденційності даних, що передаються по мережі.

– аутентифікація та авторизація:

AAA (Authentication, Authorization, and Accounting): Встановлення механізмів аутентифікації, авторизації та обліку, які контролюють доступ користувачів до мережевих ресурсів та відстежують їхню активність.

– оновлення та патчі безпеки:

Регулярне оновлення програмного забезпечення та встановлення патчів безпеки для усунення відомих уразливостей.

– захист від вірусів та інших загроз:

Використання антивірусного забезпечення для виявлення та блокування шкідливого програмного забезпечення.

– мережеві політики та процедури:

Встановлення правил та процедур для користувачів, які використовують мережу, включаючи паролі, права доступу, заборонені дії та інші аспекти безпеки.

– моніторинг та виявлення загроз:

Системи моніторингу та виявлення відомих та невідомих загроз, такі як поведінковий аналіз та аналіз журналів подій.

Ці заходи узгоджені між собою для створення комплексної системи безпеки мережі, яка мінімізує ризики та забезпечує надійний захист від різних загроз.

– стратегії реагування на інциденти безпеки.

Стратегії реагування на інциденти безпеки є критичною складовою будь-якої безпечної системи. Розглянемо детальніше основні аспекти цих стратегій:

– плани та процедури:

Визначення планів дій та процедур для реагування на різні типи інцидентів безпеки.

Розроблення чітких інструкцій щодо того, як діяти в разі виникнення інциденту, включаючи збір інформації, повідомлення зацікавленим сторонам та відновлення систем.

– виявлення інцидентів:

Використання систем моніторингу та виявлення вразливостей для раннього виявлення потенційних інцидентів безпеки.

Встановлення систем аналізу журналів подій та алертів, які сповіщають адміністраторів про підозрілу або незвичну активність.

– аналіз інцидентів:

Проведення детального аналізу виявлених інцидентів для визначення їхнього обсягу, впливу та причин.

Встановлення механізмів відстеження та документування подій під час інциденту для подальшого аналізу та уроків, які можна вивчити.

– реагування на інциденти:

Вжиття необхідних заходів для мінімізації збитків та відновлення безпеки системи [7].

Використання технік інцидентного реагування для управління виниклими ситуаціями та відновлення нормального стану системи.

– комунікація та співпраця:

Встановлення ефективної системи комунікації з усіма зацікавленими сторонами, включаючи керівництво, інформаційні служби, клієнтів та інших сторін.

Співпраця зі сторонніми експертами та агентствами з метою швидкого та ефективного реагування на складні інциденти безпеки.

– оцінка та вдосконалення:

Проведення оцінки після інциденту для визначення ефективності заходів реагування та ідентифікації можливих покращень.

Внесення коректив до планів та процедур реагування на основі вивчених уроків та рекомендацій.

Ці стратегії забезпечують гнучкість та готовність до відповіді на різні сценарії інцидентів безпеки, зменшуючи вплив на діяльність організації та забезпечуючи швидке відновлення нормального стану.

У ході аналізу критичних елементів захисту інформації було виявлено, що ці аспекти є фундаментальними для забезпечення безпеки та конфіденційності даних у будь-якій організації. Ідентифікація і аутентифікація, контроль доступу, шифрування даних, моніторинг та аудит, фізична безпека, заходи безпеки мережі та стратегії реагування на інциденти є ключовими складовими частинами сучасних систем захисту інформації [8].

Використання та належна реалізація цих елементів дозволяють забезпечити високий рівень захищеності даних від різноманітних загроз, включаючи несанкціонований доступ, втрату, крадіжку, витік чи порушення конфіденційності. Застосування комплексного підходу до захисту інформації забезпечує надійну оборону від внутрішніх та зовнішніх загроз, що може має вирішальне значення для успішної діяльності організації та збереження її репутації.

Отже, ретельна увага до критичних елементів захисту інформації є важливою передумовою для створення стійкої та безпечної інформаційної інфраструктури, що відповідає сучасним вимогам та стандартам безпеки.

### 1.3 Важливість використання резервних каналів зв’язку

В сучасному світі, де різноманітні види комунікації стали невід'ємною частиною нашого повсякденного життя, важливість використання резервних каналів зв'язку виходить на передній план. Це особливо актуально в умовах небезпеки або непередбачених ситуацій, коли доступність комунікаційних засобів може бути критичною для забезпечення безпеки, координації дій та надання допомоги [9].

Резервні канали зв'язку - це альтернативні способи комунікації, які використовуються, коли основний канал зв'язку недоступний.

Існує багато видів резервних каналів зв'язку, таких як:

– мобільний зв'язок: Мобільні телефони можуть використовуватися для голосового зв'язку, текстових повідомлень та доступу до Інтернету;

– супутниковий зв'язок: Супутниковий зв'язок може використовуватися для голосового зв'язку, текстових повідомлень та доступу до Інтернету в районах, де недоступний мобільний зв'язок;

– радіо: Радіозв'язок може використовуватися для голосового зв'язку в районах, де недоступний мобільний зв'язок або супутниковий зв'язок;

– сигнали про лихо: Сигнали про лихо, такі як димові шашки або сигнальні пістолети, можуть використовуватися для подачі сигналу про допомогу в надзвичайних ситуаціях;

– мережі радіочастотного зв'язку: Мережі радіочастотного зв'язку, такі як DMR, можуть використовуватися для голосового зв'язку та передачі даних;

– інтернет: Інтернет може використовуватися для голосового зв'язку, текстових повідомлень, електронної пошти та доступу до інформації;

– соціальні мережі: Соціальні мережі можуть використовуватися для зв'язку з людьми та організаціями;

– месенджери: Месенджери, такі як WhatsApp, Telegram, Signal, можуть використовуватися для текстового та голосового зв'язку, а також для обміну файлами.

Резервні канали зв'язку є важливою складовою систем безпеки та контингентного планування в умовах, коли основні канали зв'язку можуть бути недоступними або пошкодженими в результаті надзвичайних подій, таких як природні катастрофи, техногенні аварії або кібератаки. Використання резервних каналів зв'язку забезпечує надійність зв'язку та можливість ефективного управління ситуацією в умовах кризи.

Використання резервних каналів зв'язку має ряд переваг:

– підвищення стійкості: Забезпечує безперервність зв'язку у випадку збоїв або відмов основного каналу. Наприклад, якщо під час стихійного лиха мобільні мережі перевантажені, люди можуть використовувати радіо для зв'язку з екстреними службами;

– підвищення надійності: Зменшує ризик втрати інформації або затримки у комунікації. Наприклад, якщо під час ділової поїздки ви втратите доступ до Інтернету, ви можете використовувати мобільний зв'язок для надсилання важливих електронних листів;

– підвищення гнучкості: Дозволяє використовувати різні способи зв'язку залежно від ситуації. Наприклад, ви можете використовувати телефон для зв'язку з друзями, а електронну пошту - для зв'язку з колегами;

– підвищення безпеки: Забезпечує альтернативний канал зв'язку у випадку компрометації основного каналу. Наприклад, якщо хакери зламують ваш телефон, ви можете використовувати радіо для зв'язку з банком.

Враховуючи ці переваги, важливо мати належно підготовлені резервні канали зв'язку, які можуть бути активовані в разі надзвичайних ситуацій і забезпечити неперервність діяльності та безпеку інформаційних систем [10].

Резервні канали зв'язку можуть використовуватися в різних сферах, таких як:

– бізнес: для забезпечення безперервності роботи в разі збоїв основних каналів зв'язку;

– урядові організації: для забезпечення зв'язку з екстреними службами та населенням у надзвичайних ситуаціях;

– військові: для забезпечення зв'язку з військовослужбовцями та підрозділами в польових умовах;

– охорона здоров'я: для забезпечення зв'язку з медичними працівниками та пацієнтами в надзвичайних ситуаціях;

– освіта: для забезпечення зв'язку з учнями та викладачами в разі збоїв основних каналів зв'язку.

Резервні канали зв'язку важливі для забезпечення неперервності діяльності в умовах надзвичайних ситуацій або в разі відмови основних мереж або систем зв'язку. Вони можуть бути різних типів і технологій, що забезпечує гнучкість і надійність зв'язку в різних ситуаціях.

Основні аспекти резервних каналів зв'язку включають:

– різноманітність технологій: Резервні канали можуть використовувати різноманітні технології, такі як супутниковий зв'язок, бездротові мережі, радіочастотний зв'язок тощо. Це дозволяє підібрати найбільш підходящий та надійний засіб зв'язку залежно від умов і потреб;

– стійкість до відмов та атак: Резервні канали зв'язку можуть забезпечити стійкість до відмов і атак, оскільки вони можуть працювати незалежно від основних мереж та мати власні заходи безпеки;

– резервування пропускної здатності: Деякі резервні канали можуть бути призначені для резервування пропускної здатності, тобто вони можуть активуватися тільки у разі необхідності або в разі відмови основних каналів;

– швидкість активації: Резервні канали повинні мати можливість швидко активуватися в разі відмови основних мереж або виникнення надзвичайних ситуацій;

– плани відновлення після кризи: Важливо мати плани відновлення роботи зв'язку після виникнення надзвичайних ситуацій або відмови основних мереж;

– тестування та тренування: Резервні канали зв'язку повинні регулярно перевірятися і тестуватися для забезпечення їхньої надійності та ефективності.

Загальна мета використання резервних каналів зв'язку полягає в тому, щоб забезпечити неперервність зв'язку із зовнішнім світом навіть у найекстремальніших ситуаціях, забезпечуючи таким чином безпеку і функціонування систем та організацій.

### 1.4 Методи шифрування даних та їх важливість

Методи шифрування даних є ключовим елементом в сфері кібербезпеки та захисту інформації. Вони дозволяють перетворити звичайний текст або дані в зашифровану форму, що робить їх незрозумілими для несанкціонованого доступу. Важливість шифрування полягає в забезпеченні конфіденційності, цілісності та доступності інформації, а також в захисті від різних видів кіберзлочинності. Далі будуть розглянуті різні методи шифрування та їхні особливості [12].

Шифрування даних - це процес перетворення звичайного тексту або інформації в зашифровану форму за допомогою спеціального алгоритму (шифру), що робить дані незрозумілими для незаконного доступу. Шифрування даних є ключовим елементом для забезпечення конфіденційності, цілісності та доступності інформації. Ось детальніше про методи шифрування та їхню важливість:

– симетричне шифрування;

Симетричне шифрування - це метод, в якому для шифрування та розшифрування використовується один і той же ключ. Основна ідея полягає в тому, що інформація шифрується за допомогою ключа, і той самий ключ використовується для розшифрування її отримувачем.

Однією з важливих переваг симетричного шифрування є його ефективність та швидкість обробки великої кількості даних. Алгоритми симетричного шифрування, такі як AES (Advanced Encryption Standard), DES (Data Encryption Standard) або Triple DES, зазвичай добре оптимізовані та працюють швидко.

Проте одним з основних недоліків цього методу є потреба в безпечному обміні ключами між сторонами. Це означає, що обидва співрозмовники повинні

мати доступ до одного й того ж ключа перед взаємною комунікацією, що може бути складним завданням, особливо в великих мережах або на відкритих каналах зв'язку.

Ще однією проблемою є безпека ключів: якщо ключ буде скомпрометований, всі дані, зашифровані за його допомогою, також будуть вразливі. Тому забезпечення безпеки ключів - це важлива складова симетричного шифрування.

– асиметричне шифрування;

Асиметричне шифрування - це метод, що використовує два ключі: публічний та приватний. Публічний ключ призначений для шифрування повідомлень, а приватний - для їхнього розшифрування. Основна відмінність від симетричного шифрування полягає в тому, що у асиметричному використовуються різні ключі для шифрування та розшифрування даних, що робить його більш безпечним.

Важливість асиметричного шифрування виявляється у декількох аспектах:

– розв'язання проблеми обміну ключами: Одним з основних викликів симетричного шифрування є потреба в безпечному обміні ключами між сторонами. Асиметричне шифрування дозволяє уникнути цієї проблеми, оскільки кожен користувач має свій унікальний набір ключів;

– створення цифрових підписів: Приватний ключ може бути використаний для створення цифрового підпису, який підтверджує автентичність повідомлення або документа. Це дозволяє перевірити, чи була інформація підписана відповідним користувачем і не була змінена під час передачі;

– безпека комунікації: Асиметричне шифрування забезпечує високий рівень безпеки комунікації, оскільки навіть якщо зловмисник отримає публічний ключ, він не зможе розшифрувати повідомлення без приватного ключа;

– широке використання у криптографії: Асиметричне шифрування широко використовується у багатьох криптографічних протоколах, таких як TLS/SSL для захисту мережевого зв'язку та PGP для захисту електронної пошти.

Загалом, асиметричне шифрування вирішує багато проблем, пов'язаних із безпекою комунікації та захистом даних, і є важливим інструментом у сучасній криптографії [14].

– хеш-функції;

Хеш-функції - це криптографічні алгоритми, які приймають вхідні дані будь-якої довжини і генерують фіксований хеш-код фіксованої довжини, який є унікальним для кожного введеного набору даних. Ці функції мають кілька важливих властивостей і застосувань:

– перевірка цілісності даних: Однією з основних важливих властивостей хеш-функцій є здатність перевіряти цілісність даних. При невеликих змінах вхідних даних хеш-код, який генерується, змінюється катастрофічно, що робить неможливим відновлення вихідних даних з хеш-коду. Таким чином, вони використовуються для перевірки, чи були змінені дані під час передачі або зберігання;

– створення цифрових підписів: Хеш-функції також використовуються для створення цифрових підписів. При цьому підписувач використовує свій приватний ключ для створення хеш-коду вихідних даних, який потім може бути перевірений з відкритим ключем для підтвердження автентичності;

– генерація випадкових ключів: Хеш-функції можуть бути використані для генерації випадкових ключів, які використовуються у криптографії для шифрування даних та створення цифрових підписів;

– застосування в хеш-таблицях: Хеш-функції також використовуються в алгоритмах хеш-таблиць для ефективного пошуку та зберігання даних;

– застосування в паролях: У веб-додатках хеш-функції використовуються для збереження паролів у безпечному вигляді. Замість зберігання паролів у відкритому вигляді, вони перетворюються на хеш-коди, що ускладнює можливість отримання вихідних паролів з бази даних.

У цілому, хеш-функції є невід'ємною частиною криптографії та безпеки даних, оскільки вони забезпечують надійну перевірку цілісності даних, створюють цифрові підписи та забезпечують безпечне збереження паролів та ключів.

– гібридне шифрування.

Гібридне шифрування - це метод шифрування даних, який комбінує в собі переваги симетричного і асиметричного шифрування для забезпечення ефективності та безпеки процесу. Ось детальніше про цей підхід:

– комбінація методів шифрування: Гібридне шифрування використовує симетричне шифрування для швидкого та ефективного шифрування даних, а також асиметричне шифрування для безпечного обміну секретним ключем;

– симетричне шифрування для шифрування даних: Початкові дані шифруються за допомогою симетричного ключа, який швидкий та ефективний для обробки великих обсягів даних;

– асиметричне шифрування для обміну ключами: Перед використанням симетричного шифрування ключ потрібно обмінятиміж сторонами. Для цього використовується асиметричне шифрування, де одна сторона використовує свій публічний ключ для шифрування секретного ключа, а інша - приватний ключ для розшифрування;

– ефективність та безпека: Гібридне шифрування поєднує в собі швидкість симетричного шифрування з безпекою асиметричного. Він дозволяє безпечно передавати секретні ключі за допомогою асиметричного шифрування, тоді як сами дані можуть бути швидко зашифровані та розшифровані за допомогою симетричного шифрування;

– застосування: Гібридне шифрування широко використовується в сучасних криптографічних системах, таких як захищені протоколи обміну даними в Інтернеті (наприклад, SSL / TLS), а також в захищених комунікаційних системах та інших застосунках, де важливо забезпечити як швидкість, так і безпеку обміну даними.

Методи шифрування даних відіграють важливу роль у забезпеченні конфіденційності, цілісності та доступності інформації в сучасних системах. Симетричне шифрування забезпечує швидке та ефективне захоплення великої кількості даних, в той час як асиметричне шифрування дозволяє безпечно обмінюватися ключами і створювати цифрові підписи для перевірки автентичності

повідомлень. Хеш-функції використовуються для перевірки цілісності даних та створення цифрових підписів, а гібридне шифрування поєднує в собі переваги симетричного та асиметричного шифрування [16].

Використання цих методів дозволяє забезпечити надійний захист інформації від несанкціонованого доступу, зміни та втрати. Вони важливі для безпеки електронних транзакцій, захисту конфіденційної інформації, забезпечення цілісності даних та забезпечення довіри в онлайн-комунікаціях. Дотримання стандартів шифрування даних є важливою складовою для будь-якої організації, що працює з конфіденційною інформацією, та визначається як ключовий елемент в IT-стратегіях безпеки.

### 1.5 Аналіз видів зв’язків для обміну інформацією

У сучасному світі ефективне оповіщення населення про надзвичайні ситуації та небезпеку вимагає використання різноманітних каналів зв'язку. Аналіз різних видів зв'язків для обміну інформацією є ключовим етапом в розробці та впровадженні систем оповіщення населення. Вступ до цієї теми передбачає дослідження різноманітних засобів сповіщення, таких як мобільні додатки, SMS, електронна пошта, соціальні мережі, а також традиційні методи, такі як публічні табло, гучномовці, радіо та телевізійне мовлення. Дослідження різних типів зв'язку допоможе розробити оптимальну стратегію оповіщення, яка забезпечить швидке, ефективне та надійне інформування громадськості про небезпеку та надзвичайні ситуації [17].

Аналіз видів зв'язків для обміну інформацією можна проводити за такими критеріями:

– швидкість і доставка: Оцінка швидкості доставки повідомлень через різні канали зв'язку. Деякі канали можуть забезпечувати швидшу доставку, ніж інші, що є критичним у надзвичайних ситуаціях;

– надійність: Оцінка надійності кожного каналу зв'язку. Деякі канали можуть мати більшу стійкість до відмов або перешкод, що може зробити їх більш привабливими для використання;

– покриття і доступність: Аналіз доступності різних каналів зв'язку у різних регіонах та серед різних груп населення. Важливо обрати канали, які максимально охоплюють цільову аудиторію;

– вартість: Оцінка витрат на використання кожного зв'язку. Витрати можуть включати як прямі витрати на передачу повідомлень, так і витрати на підтримку та управління кожним каналом;

– зручність та доступність для користувачів: Врахування зручності та доступності для користувачів кожного каналу зв'язку. Деякі канали можуть бути більш прийнятними для користувачів через їхню звичність або простоту використання;

– безпека: Оцінка рівня безпеки кожного каналу зв'язку. Деякі канали можуть бути більш вразливими до атак або перехоплення, тому важливо обрати канали з належним рівнем захисту;

– можливості персоналізації та спрямованості: Деякі канали можуть дозволяти персоналізувати повідомлення та спрямовувати їх до конкретних аудиторій або зон, що може бути корисним у випадку надзвичайних ситуацій, що обмежуються на певну територію.

Аналіз видів зв'язків для обміну інформацією в контексті систем оповіщення населення важливий для забезпечення ефективного та швидкого реагування на надзвичайні ситуації. Для ефективного оповіщення населення про небезпеку, надзвичайні події або кризові ситуації використовуються різноманітні канали зв'язку. Ось деякі з них:

– мобільні додатки та SMS;

Мобільні додатки та SMS є ефективними засобами оповіщення населення через їх широкий охоплення та високу швидкість доставки інформації. Розглянемо їх детальніше:

Мобільні додатки:

– швидкість та надійність: Мобільні додатки можуть надсилати повідомлення безпосередньо на мобільні пристрої користувачів миттєво, забезпечуючи швидку та надійну доставку;

– персоналізація та геоприв'язка: Деякі додатки дозволяють персоналізувати повідомлення та спрямовувати їх до конкретних груп користувачів або регіонів, що дозволяє ефективно оповістити лише тих, хто потребує інформації;

– додаткові функції: Мобільні додатки можуть надавати додаткові корисні функції, такі як можливість перевірки рівня небезпеки, отримання порад щодо дій у надзвичайній ситуації тощо.

SMS (Short Message Service):

– висока доставка: SMS є одним з найбільш надійних способів доставки повідомлень, оскільки практично всі мобільні телефони підтримують цю послугу;

– широке охоплення: SMS можуть бути доставлені практично на будь-який мобільний телефон у будь-якій точці світу, що забезпечує широке охоплення аудиторії;

– простота використання: Відправлення SMS є дуже простим і доступним для користувачів будь-якого рівня технічної підготовки.

Мобільні додатки та SMS володіють великим потенціалом для швидкого та ефективного оповіщення населення про надзвичайні ситуації, забезпечуючи високу надійність, швидкість та доступність. Використання цих засобів може значно покращити реагування на небезпеку та забезпечити безпеку громадськості.

– електронна пошта;

Електронна пошта є одним з найбільш поширених та ефективних засобів для розсилки повідомлень та оповіщення населення про надзвичайні ситуації. Розглянемо детальніше важливість та особливості використання електронної пошти для цілей оповіщення:

Достовірність і оперативність:

– достовірність: Електронна пошта дозволяє надсилати повідомлення, які можна підписувати та перевіряти на автентичність, що забезпечує високу достовірність інформації;

– оперативність: Повідомлення можуть бути доставлені миттєво або за кілька хвилин, що дозволяє оперативно реагувати на надзвичайні ситуації та швидко інформувати населення.

Можливості персоналізації:

– цільове спрямування: Електронні листи можуть бути спрямовані на конкретні групи або підписників, що дозволяє надсилати інформацію лише тим, хто потребує її;

– персоналізований контент: Зміст повідомлення може бути адаптований для різних груп отримувачів, що забезпечує більшу ефективність комунікації.

Ефективність та економічність:

– швидкість доставки: Порівняно з традиційними методами, електронна пошта забезпечує швидку доставку повідомлень без значних затримок;

– вартість: В порівнянні з іншими каналами оповіщення, електронна пошта є відносно дешевим та економічним засобом комунікації.

Зручність та доступність:

– доступність: Почтові скриньки є доступними практично для кожного користувача з доступом до Інтернету, що робить електронну пошту одним з найбільш доступних засобів комунікації.

– мобільність: Можливість отримувати електронні повідомлення на мобільних пристроях дозволяє користувачам отримувати оперативну інформацію будь-де та будь-коли.

Електронна пошта володіє значним потенціалом як засіб оповіщення населення про надзвичайні ситуації через свою швидкість, ефективність, можливості персоналізації та доступність. Використання цього каналу сприяє ефективному та надійному розповсюдженню важливої інформації [20].

– соціальні мережі;

Соціальні мережі є потужним інструментом для розповсюдження інформації та оповіщення населення про надзвичайні ситуації через їх широкий охоплення та швидке поширення. Розглянемо детальніше особливості та важливість використання соціальних мереж для оповіщення:

Широкий охоплення та залучення аудиторії:

– Масовий охоплення: Соціальні мережі мають мільйони активних користувачів по всьому світу, що дозволяє широко розповсюджувати інформацію про надзвичайні ситуації.

– Залучення цільової аудиторії: Завдяки можливостям таргетування рекламних кампаній, інформація може бути спрямована на конкретні географічні райони або групи користувачів, що збільшує ефективність оповіщення.

Швидкість поширення інформації:

– Вірусний ефект: Інформація, яка стає віральною, може швидко поширюватися через спільноти та репости, що дозволяє швидко оповістити про надзвичайні події велику кількість осіб.

Багатомовність та глобальний охоплення:

– Мовна різноманітність: Соціальні мережі підтримують багато мов, що дозволяє надсилати повідомлення на різні мови та досягати різних культурних груп.

– Глобальний охоплення: Завдяки своїй доступності, соціальні мережі дозволяють оповіщати не тільки місцеве населення, а й людей по всьому світу.

Можливість взаємодії та обговорення:

– Обговорення та коментарі: Користувачі можуть взаємодіяти з опублікованим повідомленням, ділитися своїми думками та досвідом, що сприяє обговоренню надзвичайної ситуації та обміну корисною інформацією.

Можливість створення геотегів та геозонування:

– Геотегування: Інформація може бути маркована географічними тегами, що дозволяє надсилати оповіщення лише для користувачів у конкретних локаціях, де може виникнути надзвичайна ситуація.

Використання соціальних мереж для оповіщення населення про надзвичайні ситуації є важливим засобом, оскільки вони забезпечують широкий охоплення, швидке поширення інформації, можливість взаємодії та географічну специфіку. Такий підхід допомагає забезпечити оперативне та ефективне оповіщення населення у випадку надзвичайних ситуацій.

– автоматизовані голосові дзвінки;

Системи автоматизованих голосових дзвінків - це важливий інструмент масового оповіщення, який дозволяє надсилати стандартизовані голосові повідомлення або повідомлення, згенеровані в реальному часі, на велику кількість телефонних номерів. Цей метод оповіщення має декілька ключових переваг та особливостей.

Швидкість та оперативність: Однією з головних переваг є швидкість доставки повідомлень. Голосові дзвінки можуть бути доставлені майже миттєво, що робить їх ефективним інструментом для оперативного оповіщення про надзвичайні ситуації або важливі події.

– масштабність та гнучкість: системи автоматизованих голосових дзвінків дозволяють надсилати повідомлення великій кількості отримувачів одночасно. це особливо корисно в ситуаціях, коли необхідно оповістити велику аудиторію про важливу інформацію.

– персоналізація повідомлень: повідомлення можуть бути персоналізовані з урахуванням інформації про отримувачів, такої як їхні імена або конкретні деталі ситуації, що додає додаткової значущості та сприйняття повідомлення.

– забезпечення широкого охоплення: оскільки телефони є загальнодоступними інструментами, системи голосових дзвінків забезпечують широке охоплення аудиторії. це дозволяє доставляти повідомлення навіть тим, хто не має доступу до інтернету або не користується мобільними додатками.

– автоматизація та ефективність: автоматизована система дозволяє надсилати повідомлення без значних людських зусиль, що робить цей метод оповіщення високоефективним та малозатратним.

Таким чином, системи автоматизованих голосових дзвінків є важливим інструментом масового оповіщення, який дозволяє оперативно та ефективно доставляти важливу інформацію великій аудиторії.

– публічні табло та гучномовці;

Публічні табло та гучномовці є важливою складовою системи оповіщення населення під час надзвичайних ситуацій. Вони розміщені у різних публічних місцях, таких як вулиці, площі, парки, транспортні вузли, торгові центри та інші місця масового перебування людей. Основні функції цих систем включають:

– відтворення аварійних сигналів: Публічні табло та гучномовці призначені для відтворення різних аварійних сигналів або звукових сигналів, які можуть вказувати на ризик або небезпеку. Ці сигнали можуть бути різного типу, таких як попередження про природні катастрофи, терористичні загрози, пожежі, аварії або інші небезпечні ситуації;

– надання інструкцій та порад: Крім того, публічні табло та гучномовці можуть використовуватися для надання населенню інструкцій та порад щодо дій у разі надзвичайної ситуації. Це може включати вказівки щодо евакуації, місць збору, безпечних шляхів виходу, дій у разі пожежі чи інших небезпечних умов;

– поширення інформації: Публічні табло можуть бути використані для відображення текстової або графічної інформації про надзвичайну ситуацію. Це дозволяє оперативно та ефективно поширювати важливі повідомлення та інформацію серед великої кількості людей;

– покращення свідомості громадськості: Наявність публічних табло та гучномовців сприяє підвищенню свідомості громадськості про потенційні небезпеки та способи реагування на них. Вони допомагають залучити увагу та мобілізувати громадян у разі кризових ситуацій.

Отже, публічні табло та гучномовці відіграють важливу роль у системі оповіщення населення під час надзвичайних ситуацій, забезпечуючи ефективне та оперативне поширення інформації та інструкцій серед громадян.

– радіо та телевізійне мовлення;

Радіо та телевізійне мовлення відіграють ключову роль у системі оповіщення та інформування населення про надзвичайні ситуації та події. Ось детальніше про їхні можливості та важливість:

– широкий охоплення аудиторії: Радіо та телевізійні канали мають широкий охоплення аудиторії, що дозволяє ефективно та оперативно поширювати інформацію серед великої кількості людей. Це особливо важливо в ситуаціях надзвичайних подій, коли потрібно швидко досягти якомога більшої аудиторії;

– доступність для громадськості: Радіо та телевізійні пристрої є доступними для більшості населення і часто використовуються для отримання новин та інформації. Це дозволяє оперативно та ефективно поширювати надзвичайні повідомлення та інструкції серед громадськості;

– доступність в реальному часі: Радіо та телевізійне мовлення надають можливість отримання інформації в реальному часі. Це дозволяє оперативно реагувати на надзвичайні ситуації та отримувати актуальні дані про події та рекомендації для громадськості;

– довіра та авторитет: Радіо та телевізійні канали зазвичай володіють високим рівнем довіри серед громадськості. Інформація, яка походить з цих джерел, сприймається як авторитетна та достовірна, що є важливим у надзвичайних ситуаціях.

Отже, радіо та телевізійне мовлення є важливими інструментами для оповіщення та інформування населення про надзвичайні ситуації. Вони забезпечують швидку та ефективну ретрансляцію повідомлень та інструкцій, що допомагає зберегти життя та майно, а також координувати дії в умовах надзвичайних обставин.

– мережа Wi-Fi.

Мережа Wi-Fi, як одна з основних бездротових технологій, може використовуватися для розсилки повідомлень та надання інформаційних сервісів навіть у випадку надзвичайних ситуацій. Ось детальніше про це:

– розсилка повідомлень через Wi-Fi: У надзвичайних ситуаціях, коли звичайні комунікаційні канали можуть бути обмеженими або недоступними, мережа Wi-Fi може стати важливим засобом розсилки повідомлень. Наприклад, використання Wi-Fi у великих містах або громадських місцях може допомогти в швидкій розсилці повідомлень про надзвичайні ситуації;

– надання інформаційних сервісів: Мережа Wi-Fi може бути використана для надання інформаційних сервісів, таких як доступ до веб-сайтів з інструкціями щодо поведінки у надзвичайних ситуаціях, отримання новин та оновлення щодо подій, а також для доступу до електронних карт та інших корисних ресурсів;

– можливість підключення до мережі з будь-якого місця: Однією з переваг мережі Wi-Fi є можливість підключитися до неї з будь-якого місця, де є доступ до сигналу. Це дозволяє забезпечити комунікацію та надання інформаційних послуг навіть у віддалених або важкодоступних місцях, де можуть виникати надзвичайні ситуації;

– забезпечення ефективності та швидкості: Мережа Wi-Fi зазвичай надає високу швидкість передачі даних, що дозволяє ефективно та оперативно розсилати повідомлення та надавати доступ до інформаційних ресурсів.

Отже, мережа Wi-Fi може бути важливим елементом систем оповіщення та надання інформаційних послуг у випадку надзвичайних ситуацій, забезпечуючи швидку та ефективну комунікацію та доступ до інформації.

Для здійснення аналізу видів зв'язків для обміну інформацією було використано ряд критеріїв, що дозволяють оцінити кожен з каналів зв'язку з точки зору його ефективності, доступності, надійності та інших аспектів. Нижче наведена таблиця 1.1, що відображає результати аналізу за обраними критеріями.

Таблиця 1.1 - Аналіз видів зв'язків для обміну інформацією за обраними критеріями

| Вид зв'язку | Швидкість і доставка | Надійність | Покриття і доступність | Вартість | Зручність та доступність для користувачів | Безпека | Можливості персоналізації та спрямованості |
|---|---|---|---|---|---|---|---|
| Мобільні додатки та SMS | Висока | Висока | Велика | Низька | Висока | Висока | Середня |
| Електронна пошта | Помірна | Висока | Глобальна | Низька | Висока | Середня | Низька |
| Соціальні мережі | Висока | Помірна | Глобальна | Низька | Висока | Середня | Висока |
| Автоматизовані голосові дзвінки | Висока | Висока | Локальна | Середня | Висока | Висока | Низька |
| Публічні табло та гучномовці | Низька | Висока | Локальна | Середня | Висока | Висока | Середня |
| Радіо та телевізійне мовлення | Висока | Висока | Глобальна | Висока | Висока | Середня | Середня |
| Мережа Wi-Fi | Висока | Помірна | Обмежена | Висока | Висока | Середня | Висока |

Після аналізу різних видів зв'язків для обміну інформацією, можна визначити, що кожен з них має свої переваги та обмеження. Інтеграція різних каналів зв'язку може бути ефективним підходом у забезпеченні швидкого та надійного оповіщення населення в надзвичайних ситуаціях. Однак важливо враховувати вимоги безпеки, доступності та зручності для кінцевих користувачів при виборі каналів для обміну інформацією [21].

### 1.6 Висновки та постановка задачі

В рамках даної роботи проведено дослідження теоретичних основ захисту системи оповіщення населення про ракетну небезпеку в автоматичному режимі. Для цього було ретельно проаналізовано різні аспекти безпеки та ефективності систем оповіщення, які включають в себе важливість використання та різноманітність систем оповіщення, критичні елементи захисту інформації,

використання резервних каналів зв'язку, методи шифрування даних, аналіз видів зв'язків для обміну інформацією та інші. Дослідження дозволило отримати глибоке розуміння та оцінку ключових аспектів безпеки системи оповіщення, що є критичним у забезпеченні безпеки та захисту населення в умовах надзвичайних ситуацій.

Отже, на основі проведеного дослідження та аналізу було сформульовано ряд завдань для подальшої розробки:

– розробка алгоритму роботи пристрою оповіщення населення про ракетну небезпеку
– проектування системи резервних каналів обміну
– розробка схеми роботи пристрою оповіщення населення про ракетну небезпеку
– практична реалізація

# 2 СТВОРЕННЯ ЗАХИЩЕНОЇ СИСТЕМИ ОПОВІЩЕННЯ НАСЕЛЕННЯ ПРО РАКЕТНУ НЕБЕЗПЕКУ В АВТОМАТИЧНОМУ ТА «РУЧНОМУ» РЕЖИМАХ З ВИКОРИСТАННЯМ ОДНОСТОРОННЬОГО ЗВ'ЯЗКУ, РЕЗЕРВНИХ КАНАЛІВ ОБМІНУ ТА ШИФРУВАННЯМ ДАНИХ

В даному розділі буде здійснено розгляд основ створення захищеної системи оповіщення населення про ракетну небезпеку, після чого буде здійснено проектування системи резервних каналів обміну, також буде здійснена розробка схеми роботи пристрою оповіщення населення про ракетну небезпеку, після чого буде здійснена розробка алгоритму роботи пристрою оповіщення населення про ракетну небезпеку

## 2.1 Основи створення захищеної системи оповіщення населення про ракетну небезпеку

Створення захищеної системи оповіщення населення про ракетну небезпеку вимагає комплексного підходу, який забезпечує швидке, надійне та безпечне інформування населення у випадку загрози.

Першим кроком є чітке визначення цілей системи оповіщення та вимог до неї. Це включає забезпечення максимальної швидкості оповіщення, високу надійність передачі сигналу в умовах різних загроз, а також здатність системи працювати в умовах значних перешкод та атак на інфраструктуру.

Необхідно провести аналіз потенційних загроз та розробити сценарії можливих ракетних атак. Це допоможе визначити, які канали комунікації найкраще підходять для передачі сигналів оповіщення та які резервні механізми слід впровадити.

Розробка автоматичної системи виявлення та оповіщення, автоматизація процесів виявлення загрози та активації системи оповіщення знижує ризик людської помилки та забезпечує миттєву реакцію на небезпеку [22].

Аналіз існуючих загроз і розробка відповідних сценаріїв виступає як фундаментальний компонент у процесі створення ефективної системи оповіщення

населення про ракетну небезпеку. Цей аналіз вимагає всебічного вивчення потенційних джерел загроз, включаючи державні та недержавні актори, здатні використовувати ракетну техніку або безпілотні літальні апарати (БПЛА) для досягнення своїх стратегічних цілей. Серед таких загроз особливо виділяються ракетні атаки, використання БПЛА для доставки вибухових пристроїв або проведення розвідки, а також кібератаки, що можуть супроводжувати або передувати прямим ракетним ударам.

Розгляд типів ракетних загроз включає аналіз балістичних ракет, крилатих ракет та артилерійських ракетних систем, кожна з яких має свої унікальні характеристики, такі як дальність, точність та можливості маневрування, що значною мірою впливають на стратегії їх виявлення та перехоплення.

Сценарії використання цих загроз охоплюють широкий спектр можливостей від прямих військових атак на стратегічно важливі об'єкти та цивільну інфраструктуру до терористичних актів, спрямованих на сіяння хаосу та дестабілізацію суспільства. Крім того, не слід недооцінювати вплив психологічних операцій, включаючи ложні тривоги, що можуть мати за мету викликати паніку серед населення [23].

У контексті аналізу вразливостей системи оповіщення, особлива увага приділяється технічним, фізичним та людським факторам. Технічні вразливості включають слабкі місця в програмному забезпеченні та апаратному забезпеченні, які можуть бути використані зловмисниками для блокування або знищення системи оповіщення. Людські фактори, такі як помилки у прийнятті рішень або невірна інтерпретація даних, також вимагають ретельного аналізу та впровадження відповідних заходів з мінімізації ризиків. Фізичні вразливості, у свою чергу, акцентують на необхідності захисту критичних компонентів системи від прямого знищення в результаті атаки.

Детальний аналіз сценаріїв використання ракетних загроз та системи оповіщення населення про таку небезпеку є критично важливим для розуміння потенційних ситуацій, які можуть виникнути, і для планування відповідних реакцій. Сценарії можуть варіюватися від прямих військових атак до

терористичних актів, кожен з яких має унікальні характеристики та потребує специфічних заходів реагування.

Цей сценарій передбачає використання ракет військовими силами однієї країни проти іншої з метою завдання ударів по стратегічно важливим військовим або цивільним об'єктам. Прямі військові атаки можуть бути сплановані та виконані з різною масштабністю та інтенсивністю, і можуть включати використання балістичних або крилатих ракет великої дальності. Система оповіщення має бути спроектована таким чином, щоб забезпечити максимально швидке виявлення та оповіщення населення про такі атаки, дозволяючи людям вжити необхідних заходів безпеки.

У цьому контексті ракетні загрози або загрози використання БПЛА виникають від терористичних груп, які мають на меті створити стан хаосу, залякати населення або привернути увагу до своїх вимог. Терористичні акти можуть бути спрямовані на цивільні об'єкти, такі як громадські місця, інфраструктурні об'єкти або урядові будівлі. Сценарії реагування на терористичні акти вимагають високої точності виявлення загрози та ефективного спілкування з населенням для мінімізації потенційних жертв.

Таким чином, комплексний аналіз існуючих загроз та розробка детальних сценаріїв є невід'ємною частиною процесу проектування та реалізації системи оповіщення населення про ракетну небезпеку, забезпечуючи її здатність ефективно реагувати на широкий спектр потенційних загроз.

### 2.2 Проектування системи резервних каналів обміну

Проектування системи резервних каналів обміну є ключовим аспектом забезпечення стійкості критичних інформаційних систем, особливо у контексті національної безпеки та оповіщення населення про надзвичайні ситуації, такі як ракетні загрози. Розробка ефективної системи резервних каналів передбачає забезпечення надійного зв'язку навіть у випадку збоїв або цілеспрямованих атак на основну комунікаційну інфраструктуру.

Першим кроком у проектуванні є визначення вимог до системи резервних каналів, яке включає оцінку потенційних ризиків і загроз для існуючої інфраструктури зв'язку. Це передбачає аналіз можливих сценаріїв відмов або атак, що можуть призвести до збоїв у роботі основних каналів обміну.

Визначення вимог передбачає глибоке розуміння цілей системи, очікуваного функціоналу, а також обмежень, в межах яких система повинна функціонувати. Це включає аналіз потреб користувачів та інших зацікавлених сторін, щоб визначити критичні для безпеки характеристики, такі як швидкість передачі даних, надійність з'єднання, масштабованість системи та її спроможність до відновлення після збоїв. До важливих аспектів також належить забезпечення конфіденційності, цілісності та доступності інформації, що передається.

Аналіз ризиків полягає у систематичному виявленні та оцінці потенційних загроз для системи резервних каналів обміну, які можуть виникнути внаслідок технічних несправностей, природних катастроф, цілеспрямованих атак або людських помилок. Цей процес включає ідентифікацію вразливих компонентів системи, визначення потенційних джерел загроз і оцінку ймовірності та потенційного впливу реалізації кожної загрози. На основі проведеного аналізу розробляються стратегії мінімізації ризиків, які можуть включати технічні, організаційні та процедурні заходи.

Процес аналізу ризиків здійснюється через кілька ключових етапів: ідентифікацію загроз, оцінку вразливостей, визначення ймовірності настання загроз та оцінку потенційного впливу на систему. Результатом є ієрархія ризиків, яка допомагає приоритизувати зусилля щодо їх мінімізації та визначити необхідні ресурси для запобігання або пом'якшення наслідків потенційних інцидентів [24].

Враховуючи динамічний характер загроз і швидкі зміни у технологічному середовищі, визначення вимог та аналіз ризиків вимагають регулярного перегляду та оновлення. Це дозволяє адаптувати систему резервних каналів обміну до нових викликів і забезпечити її стійкість у довгостроковій перспективі.

Архітектура системи повинна бути розроблена з урахуванням принципів модульності, що дозволяє легко модифікувати або розширювати систему без

необхідності повної її перебудови. Модульний підхід сприяє також підвищенню надійності системи, оскільки збої в одному модулі не обов'язково призводять до відмови всієї системи.

Архітектура має забезпечувати можливість масштабування системи, щоб вона могла ефективно обробляти зростання обсягів інформації та кількості користувачів. Це важливо для систем, які повинні працювати в умовах, що динамічно змінюються, та підтримувати високу продуктивність незалежно від навантаження.

Забезпечення високого рівня відмовостійкості є критично важливим. Архітектура повинна передбачати використання резервних компонентів та механізмів автоматичного відновлення для забезпечення безперебійної роботи системи навіть у разі відмов окремих елементів [25].

Оскільки системи резервних каналів обміну часто обробляють критично важливу інформацію, архітектура має включати розгорнуті засоби захисту даних, включаючи шифрування, аутентифікацію, контроль доступу та захист від зовнішніх атак.

## 2.3 Розробка схеми роботи пристрою оповіщення населення про ракетну небезпеку

Розробка схеми роботи пристрою оповіщення населення про ракетну небезпеку вимагає інтеграції передових технологій і багаторівневого підходу до проектування для забезпечення швидкого, надійного та ефективного сповіщення громадян у критичних ситуаціях.

Розробимо схему пристрою, яка буде включати в себе резервні канали зв’язку (рис 2.1).

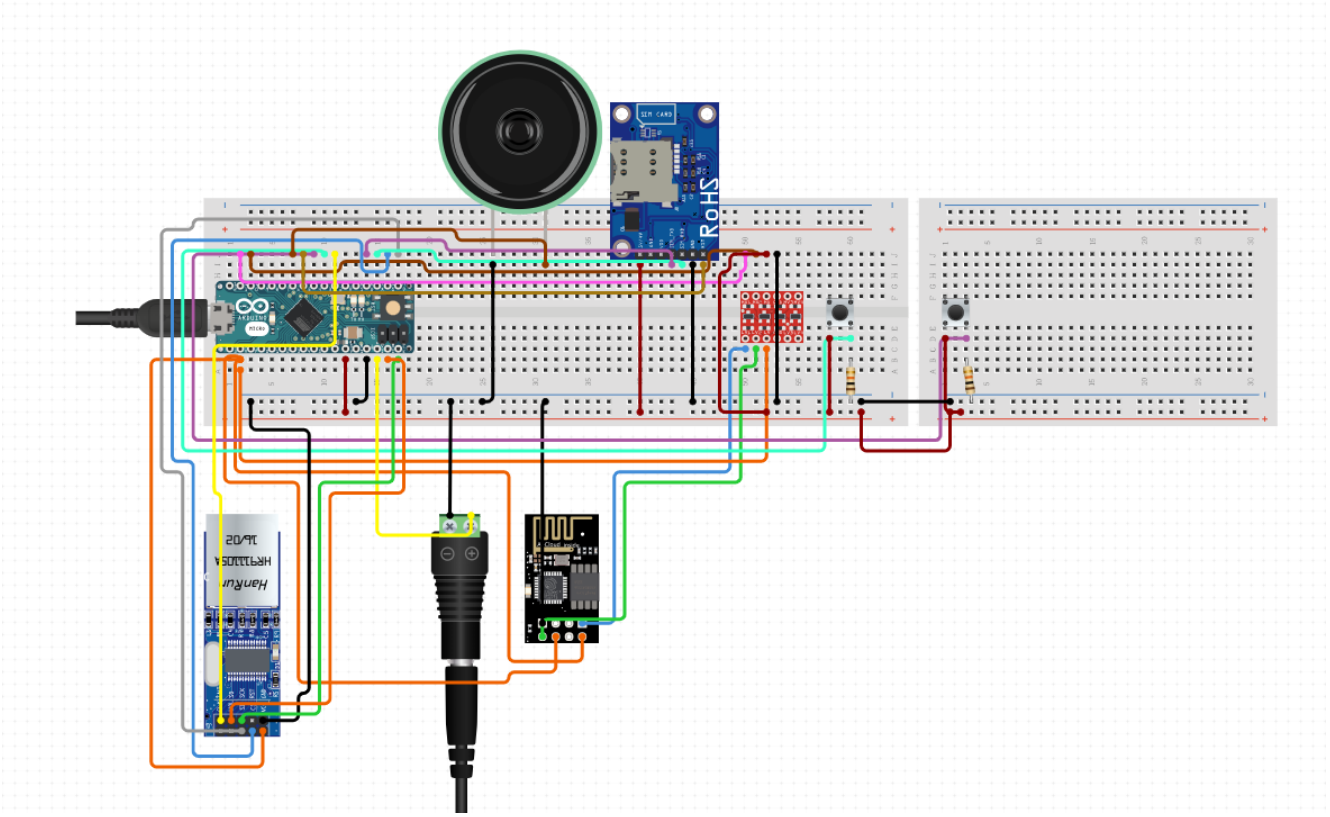


Рисунок 2.1 – Схема пристрою оповіщення населення з резервними каналами зв’язку

Така система має забезпечувати надійне сповіщення через автоматичний та "ручний" режими, використовуючи резервні канали зв'язку (GSM, WiFi, Ethernet) та шифрування даних для забезпечення безпеки інформації. Нижче представлено концептуальну схему такого пристрою:

Компоненти системи:

1. Arduino Board: Як центральний контролер, що використовується для керування процесами детекції загроз, генерації сигналів оповіщення, управління передачею даних через різні канали та обробки входів від ручного інтерфейсу.
2. Модуль GSM: Дозволяє відправляти SMS-повідомлення або здійснювати дзвінки для оповіщення населення у випадку виявлення ракетної загрози.
3. WiFi Модуль (наприклад, ESP8266): Використовується для надсилання оповіщень через інтернет, включаючи електронну пошту, push-повідомлення в мобільних додатках або повідомлення в соціальних мережах.

4. Ethernet Модуль: Забезпечує пряме підключення до мережі Internet через кабель, як резервний канал зв'язку для випадків, коли бездротові мережі недоступні.
5. Кнопки для "ручного" запуску: Дозволяють оператору системи активувати оповіщення вручну у випадку необхідності.
6. Модуль шифрування: Впроваджує алгоритми шифрування для захисту переданих повідомлень, забезпечуючи їхню конфіденційність та цілісність.

Схема роботи:

Автоматичний режим:

1. Детекція загрози: Arduino постійно отримує дані від зовнішніх систем раннього попередження через інтернет або інші доступні канали.
2. Активація сигналу: При ідентифікації загрози Arduino автоматично генерує сигнал оповіщення.
3. Вибір каналу зв'язку: Система визначає найбільш надійний канал зв'язку на даний момент (WiFi, GSM, Ethernet) і спрямовує через нього повідомлення.
4. Шифрування та передача: Повідомлення шифрується перед відправкою через обраний канал зв'язку.

"Ручний" режим:

1. Активація: Оператор активує "ручний" режим за допомогою фізичних кнопок на пристрої.
2. Вибір каналу та введення повідомлення: Оператор вибирає канал зв'язку та вводить текст повідомлення вручну або вибирає з готових шаблонів.
3. Шифрування та відправлення: Повідомлення шифрується та відправляється через обраний канал.

Така схема роботи забезпечує гнучкість та надійність системи оповіщення, дозволяючи швидко реагувати на загрози ракетного удару та ефективно інформувати населення за будь-яких умов.

## 2.4 Розробка алгоритму роботи пристрою оповіщення населення про ракетну небезпеку

Розробка алгоритму роботи пристрою оповіщення населення про ракетну небезпеку передбачає створення логічної послідовності кроків, яка дозволяє системі швидко та ефективно реагувати на загрози, забезпечуючи оперативне інформування населення. Алгоритм має враховувати автоматичне виявлення загроз, вибір каналів зв'язку для оповіщення, а також можливість "ручного" втручання для управління системою.

Розробимо та опишемо даний алгоритм (рис. 2.2).

Крок 1: Початок

- алгоритм розпочинається з ініціації системи.

Крок 2: Перевірка статусу з'єднання

- система перевіряє наявність з'єднання з мережею або з джерелом даних про ракетну загрозу.

Крок 3: Зв'язок є?

- виконується умовна перевірка, чи є з'єднання активним.

Крок 4: Відправлення запиту для перевірки статусу

- якщо з'єднання є, система відправляє запит до відповідних джерел інформації для отримання даних про статус ракетної загрози.

Крок 5: Відповідь є?

- система перевіряє, чи отримано відповідь на свій запит.

Крок 6: Аналіз відповіді

- після отримання відповіді, система аналізує надану інформацію для визначення рівня загрози.

Крок 7: Перевірка статусу зв'язку через резервні канали

- якщо зв'язок через основний канал втрачено, система переходить до використання резервних каналів: gsm, wifi або ethernet.

Крок 8: Відправлення відповіді через інший канал зв'язку


Рисунок 2.2 - Алгоритм роботи пристрою оповіщення населення про ракетну небезпеку

– система відправляє запити через резервні канали зв'язку, щоб отримати оновлену інформацію про статус загрози.

Крок 9: Відповідь однакова зі всіх каналів, де є зв'язок?

– виконується перевірка на консистентність отриманих відповідей через різні канали зв'язку.

Крок 10: Повідомлення про тривогу

– якщо аналіз відповідей підтверджує загрозу ракетного удару, система активує оповіщення населення.

Крок 11: Повернення до початку

– після активації оповіщення, або якщо загрозу не виявлено, система повертається до початкового стану для продовження моніторингу.

Цей алгоритм враховує використання багаторівневого підходу до перевірки інформації та оповіщення, включаючи автоматичну та "ручну" системи перевірки статусу загрози, а також запобіжні механізми для забезпечення надійності зв'язку через резервні канали у випадку збою основного каналу.

### 2.5 Висновки до розділу.

У процесі виконаної роботи було здійснено комплексну розробку алгоритму функціонування пристрою оповіщення населення про ракетну небезпеку. Основною метою розробки було створення багатофункціональної, надійної та гнучкої системи, здатної забезпечити швидке реагування на потенційні загрози та ефективне інформування населення в умовах різних викликів інформаційної безпеки.

Під час розробки алгоритму були враховані ключові аспекти, такі як можливість використання різноманітних комунікаційних каналів, у тому числі GSM, WiFi та Ethernet, як резервних каналів зв'язку. Система здатна працювати в автоматичному режимі, забезпечуючи оперативність реакції, а також передбачає "ручний" режим для ситуацій, що потребують індивідуального підходу.

Впровадження сучасних методів шифрування даних дозволяє забезпечити високий рівень безпеки інформації, що транслюється системою оповіщення.

Узагальнюючи проведену роботу, можна зазначити, що створена система оповіщення є високотехнологічним рішенням, орієнтованим на забезпечення національної безпеки. Архітектура системи побудована з дотриманням принципів модульності, шкалованості та відмовостійкості, що разом з механізмами шифрування даних і використанням резервних каналів зв'язку робить систему оповіщення готовою до роботи в екстремальних умовах, забезпечуючи надійність та оперативність інформування громадськості.

# 3 ПРОГРАМНА РЕАЛІЗАЦІЯ

## 3.1 Обґрунтування вибору мови програмування та середовища розробки

Мова програмування C++ була обрана для реалізації системи оповіщення населення про ракетну небезпеку з кількох вагомих причин.

– C++ є однією з найпродуктивніших мов програмування, яка забезпечує швидке виконання коду та ефективне управління пам'яттю. Це особливо важливо для систем реального часу, де швидкість реакції є критичним фактором.

– C++ дозволяє програмістам мати прямий контроль над апаратними ресурсами, що є важливим для вбудованих систем, де необхідно працювати з різними датчиками та модулями.

– C++ підтримує об'єктно-орієнтоване програмування (ООП), що дозволяє створювати модульні, масштабовані та легко підтримувані додатки. ООП підхід допомагає структурувати код у вигляді об'єктів, що взаємодіють один з одним, що спрощує розробку та тестування.

– C++ має багату стандартну бібліотеку, а також велику кількість сторонніх бібліотек, які можна використовувати для різних завдань, включаючи мережеві з'єднання, шифрування даних та роботу з апаратним забезпеченням.

– C++ дозволяє створювати додатки, які можуть бути запущені на різних операційних системах з мінімальними змінами в коді. Це забезпечує гнучкість і мобільність системи.

C++ дозволяє створювати додатки, які можуть бути запущені на різних операційних системах з мінімальними змінами в коді. Це забезпечує гнучкість і мобільність системи, оскільки вона може бути розгорнута на різних платформах, включаючи Windows, Linux та macOS. Така кросплатформеність також сприяє підвищенню надійності та стійкості системи.

Компілятори C++ оптимізовані для генерації високопродуктивного машинного коду, що забезпечує швидке виконання програм. Більшість сучасних компіляторів, таких як GCC, Clang та Microsoft Visual C++, підтримують розширені

оптимізації, що дозволяє отримати додатковий приріст продуктивності. Це особливо важливо для реальних умов експлуатації системи оповіщення, де час виконання кожної операції може мати вирішальне значення.

C++ постійно розвивається, і нові стандарти (наприклад, C++11, C++14, C++17, C++20) додають нові можливості та покращення. Це включає покращену підтримку багатопотоковості, нові контейнери та алгоритми, лямбда-функції, модулі та інші корисні функції. Використання сучасних стандартів C++ дозволяє створювати більш ефективний та зрозумілий код, який легше підтримувати та розширювати [28].

C++ підтримується багатьма популярними інструментами та середовищами розробки, що дозволяє вибрати найбільш підходящий набір інструментів для конкретного проекту. Це включає потужні IDE, такі як Visual Studio та CLion, системи контролю версій, системи автоматичного тестування та безперервної інтеграції. Така широка підтримка інструментів сприяє підвищенню продуктивності розробки та якості кінцевого продукту.

Загалом, вибір мови програмування C++ обґрунтований її високою продуктивністю, низькорівневим доступом до апаратних ресурсів, підтримкою об'єктно-орієнтованого програмування, багатою стандартною та сторонніми бібліотеками, кросплатформеністю, високою продуктивністю компіляції та виконання, підтримкою сучасних стандартів, а також широкою підтримкою інструментів та середовищ розробки.

Visual Studio є одним з найпопулярніших та найпотужніших інтегрованих середовищ розробки (IDE) для C++ і пропонує широкий спектр інструментів та функцій, що робить його оптимальним вибором для розробки системи оповіщення про ракетну небезпеку [29].

Visual Studio надає розширені засоби для відладки, включаючи можливість покрокового виконання коду, налаштування точок зупину, перегляд змінних у реальному часі, аналіз стеку викликів та багато інших функцій. Це значно спрощує процес виявлення та усунення помилок, що є критичним для розробки надійної

Visual Studio підтримує інтеграцію з різними системами контролю версій, такими як Git та Subversion. Це дозволяє ефективно керувати версіями коду, відстежувати зміни та співпрацювати в команді, що особливо важливо для великих проектів [30].

Visual Studio містить вбудовані засоби для створення та виконання тестів, що дозволяє розробникам автоматизувати процес тестування та забезпечити високу якість програмного забезпечення. Можливість написання модульних, інтеграційних та регресійних тестів дозволяє своєчасно виявляти та виправляти помилки.

Інструменти автозаповнення коду та рефакторингу в Visual Studio допомагають розробникам швидше писати код, зменшуючи кількість помилок та підвищуючи продуктивність. Інтелектуальне автозаповнення, підказки та автоматичний рефакторинг дозволяють зосередитися на логіці програми, а не на синтаксисі.

Visual Studio підтримує новітні стандарти C++, включаючи C++11, C++14, C++17 та C++20. Це дозволяє використовувати найновіші можливості мови, покращуючи продуктивність та зручність розробки.

Visual Studio має велику кількість доступних розширень та плагінів, що дозволяють налаштувати середовище розробки під конкретні потреби проекту. Це може включати додаткові інструменти для аналізу коду, підтримку різних мов програмування, інтеграцію з іншими сервісами та багато іншого [31].

Visual Studio має тісну інтеграцію з Azure DevOps, що дозволяє автоматизувати процеси CI/CD (безперервна інтеграція та безперервна доставка). Це забезпечує швидке та надійне розгортання додатків, а також спрощує управління проектами та командну роботу.

Visual Studio має розгалужену документацію та велику спільноту користувачів, що забезпечує швидкий доступ до ресурсів та допомоги. Це спрощує вирішення проблем, що виникають у процесі розробки, та дозволяє швидко знайти відповіді на питання.

Visual Studio надає розширені інструменти для роботи з базами даних, що дозволяє легко створювати, редагувати та управляти базами даних безпосередньо з середовища розробки. Це особливо корисно для серверної частини системи оповіщення, яка може вимагати зберігання та обробки великої кількості даних [32].

Visual Studio інтегрується з різними сервісами Microsoft, такими як Azure, Office 365 та інші. Це забезпечує додаткові можливості для розробки, хостингу та моніторингу додатків, що може бути корисним для забезпечення надійності та масштабованості системи оповіщення.

Visual Studio дозволяє налаштовувати інтерфейс користувача відповідно до потреб розробника. Це включає налаштування панелей інструментів, вікон, тем та шрифтів, що підвищує зручність роботи та продуктивність.

Visual Studio підтримує багато інших мов програмування, таких як C#, F#, Python, JavaScript та інші. Це забезпечує гнучкість у виборі технологій та дозволяє використовувати одне середовище розробки для різних проектів [33].

Загалом, вибір Visual Studio як середовища розробки для серверної частини системи оповіщення обґрунтований його потужними засобами відладки та аналізу коду, інтеграцією з системами контролю версій, інструментами для тестування, автозаповненням коду та рефакторингом, підтримкою сучасних стандартів C++, наявністю розширень та плагінів, інтеграцією з Azure DevOps, а також широкою документацією та підтримкою спільноти. Це робить Visual Studio оптимальним вибором для розробки надійної та ефективної системи оповіщення про ракетну небезпеку.

### 3.2 Програмна реалізація серверної частини

Реалізація серверної частини системи оповіщення про ракетну небезпеку є важливим етапом проекту, оскільки сервер виконує центральні функції, такі як обробка даних, керування оповіщеннями та забезпечення надійності та безпеки системи. Для цього використовуються такі технології та підходи, як C++ для

програмування, Visual Studio як середовище розробки та відповідні бібліотеки для забезпечення мережевої взаємодії, шифрування даних та резервного копіювання.

Серверна частина системи включає кілька ключових компонентів:

– модуль прийому та обробки даних: відповідає за прийом сигналів про небезпеку від датчиків та інших джерел, їхню первинну обробку та зберігання у базі даних.

– модуль керування оповіщеннями: визначає необхідність надсилання оповіщень на підставі отриманих даних, формує повідомлення та відправляє їх через доступні канали зв'язку.

– модуль шифрування: забезпечує шифрування даних, що передаються, для забезпечення їхньої конфіденційності та захисту від несанкціонованого доступу.

– резервний модуль: відповідає за збереження та відновлення даних у випадку збоїв або атак на серверну частину системи.

Для реалізації мережевої взаємодії серверної частини використовуються бібліотеки, що забезпечують створення та управління сокетами. Однією з найпопулярніших бібліотек для роботи з мережевими з'єднаннями в C++ є Boost.Asio. Ця бібліотека забезпечує асинхронний ввід-вивід та підтримує різні протоколи передачі даних, такі як TCP та UDP.

```
#include <boost/asio.hpp>
#include <iostream>
#include <thread>
using boost::asio::ip::tcp;
void session(tcp::socket socket) {
   try {
      while (true) {
         char data[1024];
         boost::system::error_code error;
         size_t length = socket.read_some(boost::asio::buffer(data), error);
         if (error == boost::asio::error::eof) {
            break; // Connection closed cleanly by peer.
         } else if (error) {
            throw boost::system::system_error(error); // Some other error.
         std::cout << "Received: " << std::string(data, length) << std::endl;
   } catch (std::exception& e) {
      std::cerr << "Exception in thread: " << e.what() << std::endl;
int main() {
   try {
      boost::asio::io_context io_context;
```

```
        tcp::acceptor acceptor(io_context, tcp::endpoint(tcp::v4(), 12345));
        while (true) {
            tcp::socket socket(io_context);
            acceptor.accept(socket);
            std::thread(session, std::move(socket)).detach();}
    } catch (std::exception& e) {
        std::cerr << "Exception: " << e.what() << std::endl;
    return
```

Модуль обробки даних відіграє ключову роль у системі оповіщення про ракетну небезпеку. Його основна задача – аналіз отриманих сигналів, визначення рівня загрози та формування відповідних повідомлень. Детальна реалізація цього модуля включає кілька основних етапів: обробка вхідних даних, аналіз загроз, генерація повідомлень та взаємодія з іншими модулями системи.

Клас DataProcessor є основним компонентом модуля обробки даних. Він відповідає за прийом сирих даних, їх обробку, аналіз та генерацію повідомлень. Нижче наведено структуру цього класу та його основні методи.

```
class DataProcessor {
public:
    std::string process_data(const std::string& data); // Головний метод обробки даних
private:
    std::string analyze_data(const std::string& data); // Метод аналізу даних
    std::string create_alert_message(const std::string& alert_level); // Метод генерації повідомлень
    int calculate_threat_level(const std::string& data); // Метод розрахунку рівня загрози
    bool validate_data(const std::string& data); // Метод валідації даних
};
```

Метод process_data є головним методом, який викликається для обробки вхідних даних. Він приймає на вхід сирі дані, передає їх на аналіз та створює відповідне повідомлення на основі результатів аналізу.

```
std::string DataProcessor::process_data(const std::string& data) {
    if (!validate_data(data)) {
        return "Invalid data received.";
    }
    std::string alert_level = analyze_data(data);
    return create_alert_message(alert_level);
}
```

Метод validate_data перевіряє цілісність та автентичність отриманих даних. Це може включати перевірку формату даних, контрольних сум, підписів або інших засобів валідації.

```
bool DataProcessor::validate_data(const std::string& data) {
```

```
    if (data.empty() || data.length() > 1024) {
        return false;
    }
    return true;
}
```

Метод analyze_data аналізує вхідні дані та визначає рівень загрози. Він може використовувати різні алгоритми аналізу, зокрема, пошук ключових слів, патернів або аналіз метрик.

```
std::string DataProcessor::analyze_data(const std::string& data) {
    int threat_level = calculate_threat_level(data);
    if (threat_level >= 80) {
        return "high";
    } else if (threat_level >= 50) {
        return "medium";
    } else {
        return "low";
    }
}
```

Метод calculate_threat_level розраховує рівень загрози на основі вхідних даних. Для цього може використовуватись різноманітна логіка, включаючи аналіз частоти певних подій, вагових коефіцієнтів різних параметрів тощо.

```
int DataProcessor::calculate_threat_level(const std::string& data) {
    int threat_level = 0;
    if (data.find("missile") != std::string::npos) {
        threat_level += 50; }
    if (data.find("alert") != std::string::npos) {
        threat_level += 30;}
    return threat_level;}
```

Метод create_alert_message генерує повідомлення на основі рівня загрози, визначеного методом analyze_data.

```
std::string DataProcessor::create_alert_message(const std::string& alert_level) {
    if (alert_level == "high") {
        return "Missile threat detected! Take immediate action!";
    } else if (alert_level == "medium") {
        return "Potential threat detected. Stay alert.";
    } else {
        return "No immediate threat detected.";
    }
}
```

Модуль обробки даних взаємодіє з іншими модулями системи, забезпечуючи їх необхідною інформацією для подальшої обробки. Наприклад:

– передає результати аналізу до модуля управління базою даних для збереження історії подій

– передає сформовані повідомлення до модуля генерації оповіщень для створення кінцевих повідомлень

– взаємодіє з модулем передачі оповіщень для доставки повідомлень кінцевим користувачам

Модуль захисту даних забезпечує шифрування, автентифікацію та цілісність даних, що передаються, зберігаються та обробляються в системі оповіщення про ракетну небезпеку. Реалізація цього модуля включає використання криптографічних бібліотек, таких як OpenSSL, для надання необхідного рівня безпеки. Нижче детально розглянуто основні аспекти реалізації модуля захисту даних.

Клас CryptoManager є основним компонентом модуля захисту даних. Він відповідає за шифрування, дешифрування, автентифікацію та перевірку цілісності даних. Нижче наведено структуру цього класу та його основні методи.

```
#include <openssl/evp.h>
#include <openssl/aes.h>
#include <openssl/rand.h>
#include <openssl/hmac.h>
#include <iostream>
#include <string>
#include <vector>
class CryptoManager {
public:
    CryptoManager();
    ~CryptoManager();
    std::vector<unsigned char> encrypt(const std::string& plaintext);
    std::string decrypt(const std::vector<unsigned char>& ciphertext);
    std::vector<unsigned char> generate_hmac(const std::string& data);
    bool verify_hmac(const std::string& data, const std::vector<unsigned char>& hmac);
private:
    EVP_CIPHER_CTX* encrypt_ctx;
    EVP_CIPHER_CTX* decrypt_ctx;
    unsigned char key[32]; // 256-bit key
```

```
    unsigned char iv[16];  // 128-bit IV
    unsigned char hmac_key[32]; // 256-bit HMAC key
    void initialize_keys();
};
```

Конструктор класу CryptoManager ініціалізує контексти шифрування та дешифрування, а також генерує ключі для шифрування та HMAC.

```
CryptoManager::CryptoManager() {
    encrypt_ctx = EVP_CIPHER_CTX_new();
    decrypt_ctx = EVP_CIPHER_CTX_new();
    initialize_keys();
}
CryptoManager::~CryptoManager() {
    EVP_CIPHER_CTX_free(encrypt_ctx);
    EVP_CIPHER_CTX_free(decrypt_ctx);
}
void CryptoManager::initialize_keys() {
    // Генерація ключа та IV для AES
    RAND_bytes(key, sizeof(key));
    RAND_bytes(iv, sizeof(iv));
    // Генерація ключа для HMAC
    RAND_bytes(hmac_key, sizeof(hmac_key));
}
```

Метод encrypt шифрує вхідний текст за допомогою алгоритму AES-256-CBC.

```
std::vector<unsigned char> CryptoManager::encrypt(const std::string& plaintext) {
    std::vector<unsigned char> ciphertext(plaintext.size() + AES_BLOCK_SIZE);
    int len;
    int ciphertext_len;
    EVP_EncryptInit_ex(encrypt_ctx, EVP_aes_256_cbc(), NULL, key, iv);
    EVP_EncryptUpdate(encrypt_ctx, ciphertext.data(), &len, reinterpret_cast<const unsigned
char*>(plaintext.c_str()), plaintext.length());
    ciphertext_len = len;
    EVP_EncryptFinal_ex(encrypt_ctx, ciphertext.data() + len, &len);
    ciphertext_len += len;

    ciphertext.resize(ciphertext_len);
    return ciphertext;
}
```

Метод decrypt дешифрує шифротекст за допомогою алгоритму AES-256-CBC.

```
std::string CryptoManager::decrypt(const std::vector<unsigned char>& ciphertext) { std::vector<unsigned char>
plaintext(ciphertext.size()); int len; int plaintext_len; EVP_DecryptInit_ex(decrypt_ctx, EVP_aes_256_cbc(), NULL, key, iv);
EVP_DecryptUpdate(decrypt_ctx, plaintext.data(), &len, ciphertext.data(), ciphertext.size()); plaintext_len = len;
```

```
EVP_DecryptFinal_ex(decrypt_ctx, plaintext.data() + len, &len); plaintext_len += len; plaintext.resize(plaintext_len); return std::string(plaintext.begin(), plaintext.end()); }
```

Метод generate_hmac генерує HMAC для даних, що забезпечує їх автентифікацію та цілісність.

```
std::vector<unsigned char> CryptoManager::generate_hmac(const std::string& data) {
  unsigned char* hmac_result;
  unsigned int len;
  hmac_result = HMAC(EVP_sha256(), hmac_key, sizeof(hmac_key), reinterpret_cast<const unsigned char*>(data.c_str()), data.length(), NULL, &len);
  return std::vector<unsigned char>(hmac_result, hmac_result + len);
}
```

Модуль захисту даних взаємодіє з іншими модулями системи, забезпечуючи шифрування, автентифікацію та перевірку цілісності даних. Наприклад:

– модуль прийому даних використовує CryptoManager для дешифрування вхідних даних

– модуль передачі оповіщень використовує CryptoManager для шифрування повідомлень перед їх відправленням

– модуль управління базою даних використовує CryptoManager для шифрування та дешифрування даних перед їх збереженням та доступом

Модуль управління базою даних (БД) відіграє важливу роль у зберіганні та управлінні даними, необхідними для функціонування системи оповіщення про ракетну небезпеку. Цей модуль відповідає за збереження історії подій, інформації про користувачів, налаштувань системи та інших важливих даних. Реалізація цього модуля включає використання реляційної бази даних, наприклад, MySQL або PostgreSQL, та бібліотеки для роботи з БД, такі як MySQL Connector/C++ або libpqxx для PostgreSQL.

Клас DatabaseManager є основним компонентом модуля управління базою даних. Він відповідає за підключення до БД, виконання SQL-запитів та обробку результатів. Нижче наведено структуру цього класу та його основні методи.

```
#include <mysql_driver.h>
#include <mysql_connection.h>
#include <cppconn/driver.h>
```

```
#include <cppconn/exception.h>
#include <cppconn/resultset.h>
#include <cppconn/statement.h>
#include <cppconn/prepared_statement.h>
#include <string>
#include <vector>
class DatabaseManager {
public:
  DatabaseManager(const std::string& db_host, const std::string& db_user, const std::string& db_password, const std::string& db_name);
  ~DatabaseManager();
  void connect();
  void disconnect();
  void insert_event(const std::string& event_type, const std::string& event_description);
  std::vector<std::pair<std::string, std::string>> get_events();
private:
  sql::Driver* driver;
  sql::Connection* connection;
  std::string db_host;
  std::string db_user;
  std::string db_password;
  std::string db_name;
};
```

Конструктор класу DatabaseManager ініціалізує параметри підключення до БД, а деструктор забезпечує коректне закриття підключення.

```
DatabaseManager::DatabaseManager(const std::string& db_host, const std::string& db_user, const std::string& db_password, const std::string& db_name)
  : db_host(db_host), db_user(db_user), db_password(db_password), db_name(db_name), connection(nullptr) {
  driver = get_driver_instance();
}
DatabaseManager::~DatabaseManager() {
  disconnect();
}
```

Модуль управління базою даних взаємодіє з іншими модулями системи, забезпечуючи збереження та доступ до необхідних даних. Наприклад:

- модуль обробки даних використовує DatabaseManager для збереження історії подій та результатів аналізу

- модуль управління користувачами використовує DatabaseManager для збереження інформації про користувачів та їхні налаштування
- модуль управління налаштуваннями використовує DatabaseManager для збереження та доступу до конфігурацій системи

### 3.3 Програмна реалізація апаратної частини

Апаратна частина системи оповіщення про ракетну небезпеку складається з розробки програмного забезпечення для контролю та управління фізичними пристроями, такими як сирени, світлові сигнали та інші оповіщувачі. Використовується платформа Arduino, що дозволяє легко керувати електронними компонентами та забезпечувати надійну роботу системи в автоматичному та ручному режимах, з одностороннім зв’язком без відповіді назад та резервними каналами обміну даними.

Основні компоненти апаратної частини

- контролер Arduino – центральний процесор, який керує всіма підключеними пристроями.
- сирена – аудіооповіщувач, що активується при виявленні загрози.
- світловий сигнал – візуальний оповіщувач, що використовується для попередження населення.
- комунікаційні модулі – для зв'язку з серверною частиною та отримання команд.
- резервні канали обміну – додаткові засоби зв’язку для забезпечення надійності системи.

Програмне забезпечення для Arduino розробляється з використанням Arduino IDE та написане мовою програмування C++. Основні компоненти програмного коду включають:

- налаштування апаратних інтерфейсів
- отримання даних від серверної частини через основний та резервний канали

– керування сиренами та світловими сигналами

– обробку команд в автоматичному та ручному режимах

```
#include <SPI.h>
#include <Ethernet.h>
// Встановлення апаратних пінів для сирени та світлового сигналу
const int sirenPin = 8;
const int lightPin = 9;
// MAC-адреса та IP-адреса Ethernet щита
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192, 168, 1, 177);
IPAddress server(192, 168, 1, 2); // IP-адреса сервера
EthernetClient client;
void setup() {
  // Налаштування пінів
  pinMode(sirenPin, OUTPUT);
  pinMode(lightPin, OUTPUT);
  // Ініціалізація Ethernet з'єднання
  Ethernet.begin(mac, ip);
  Serial.begin(9600);
  // Затримка для стабілізації з'єднання
  delay(1000);
}
void loop() {
  // Підключення до сервера
  if (client.connect(server, 80)) {
    // Запит до сервера для отримання команди
    client.println("GET /get_command HTTP/1.1");
    client.println("Host: 192.168.1.2");
    client.println("Connection: close");
    client.println();
    // Очікування відповіді від сервера
    while (client.connected()) {
      if (client.available()) {
        String line = client.readStringUntil('\n');
        if (line == "\r") {
          break;
        }
      }
    }
    // Отримання команди від сервера
    String command = client.readStringUntil('\n');
    Serial.println("Received command: " + command);
    // Виконання команди в залежності від режиму
    if (command == "ACTIVATE_SIREN") {
      digitalWrite(sirenPin, HIGH);  // Активувати сирену
    } else if (command == "DEACTIVATE_SIREN") {
      digitalWrite(sirenPin, LOW);   // Деактивувати сирену
    } else if (command == "ACTIVATE_LIGHT") {
      digitalWrite(lightPin, HIGH);  // Активувати світловий сигнал
    } else if (command == "DEACTIVATE_LIGHT") {
      digitalWrite(lightPin, LOW);   // Деактивувати світловий сигнал
    }
    // Закриття з'єднання з сервером
    client.stop();
  }
```

```
  // Перевірка резервних каналів (якщо основний канал недоступний)
  // Реалізація роботи з резервними каналами може включати додаткові модулі зв’язку
  // Затримка перед наступною ітерацією
  delay(1000);
}
```

1. налаштування апаратних пінів та мережі
   - pinMode(sirenPin, OUTPUT) та pinMode(lightPin, OUTPUT) встановлюють піні для сирени та світлового сигналу в режим виходу.
   - Ethernet.begin(mac, ip) ініціалізує Ethernet з'єднання.
2. отримання команд від серверної частини
   - Підключення до сервера за допомогою client.connect(server, 80).
   - Надсилання запиту до сервера для отримання команди.
   - Читання відповіді сервера та збереження команди.
3. виконання команд в автоматичному та ручному режимах
   - активізація або деактивація сирени та світлового сигналу на основі отриманої команди.
4. резервні канали обміну даними
   - код може бути розширено для використання резервних каналів зв'язку, таких як GSM, радіозв’язок або інші модулі для забезпечення надійності системи.

2. Взаємодія з іншими модулями

Апаратна частина взаємодіє з серверною частиною системи для отримання команд та надсилання зворотньої інформації. Зокрема:

– серверна частина передає команди на активацію або деактивацію сирен та світлових сигналів

– апаратна частина отримує команди через основний та резервні канали обміну даними

– апаратна частина реалізує односторонній зв'язок, виконуючи команди без зворотного зв’язку до сервера

Програмна реалізація апаратної частини системи оповіщення про ракетну небезпеку включає розробку програмного забезпечення для контролю та управління фізичними пристроями на платформі Arduino. Використання Arduino та

відповідних бібліотек дозволяє забезпечити надійне функціонування системи, швидке реагування на загрози та ефективне оповіщення населення. Односторонній зв'язок, резервні канали обміну даними та підтримка автоматичного та ручного режимів забезпечують надійність та гнучкість системи.

Модуль резервних каналів даних є ключовим компонентом системи оповіщення про ракетну небезпеку, що забезпечує надійність та стійкість зв'язку у разі виходу з ладу основного каналу. Модуль призначений для підтримки зв'язку між апаратною частиною та сервером через альтернативні засоби зв'язку, такі як GSM, радіозв'язок або інші мережеві модулі.

Програмне забезпечення для модуля резервних каналів розробляється з використанням Arduino IDE та написане мовою програмування C++. Основні компоненти програмного коду включають:

- налаштування основного та резервних каналів зв'язку
- моніторинг стану з'єднання
- автоматичне перемикання на резервний канал у разі збоїв
- забезпечення безпеки даних

```
#include <SPI.h>
#include <Ethernet.h>
#include <SoftwareSerial.h>
const int sirenPin = 8;
const int lightPin = 9;
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192, 168, 1, 177);
IPAddress server(192, 168, 1, 2); // IP-адреса сервера
SoftwareSerial gsmSerial(10, 11); // RX, TX піні для GSM модуля
const char* apn = "your_apn"; // APN для підключення до мобільного інтернету
const char* gprsUser = "user"; // ім'я користувача
const char* gprsPass = "password"; // пароль
EthernetClient ethernetClient;
bool isMainChannelActive = true;

void setup() {
  pinMode(sirenPin, OUTPUT);
  pinMode(lightPin, OUTPUT);
```

```
  // Ініціалізація Ethernet з'єднання
  Ethernet.begin(mac, ip);
  Serial.begin(9600);
  // Ініціалізація GSM модуля
  gsmSerial.begin(9600);
  initializeGSM();
  // Затримка для стабілізації з'єднання
  delay(1000);
}
void loop() {
  if (isMainChannelActive) {
    if (!connectToServer(ethernetClient)) {
      Serial.println("Main channel failed, switching to backup.");
      isMainChannelActive = false;
  // Затримка перед наступною ітерацією
  delay(1000);
}
bool connectToServer(Client &client) {
  if (client.connect(server, 80)) {
    // Запит до сервера для отримання команди
    client.println("GET /get_command HTTP/1.1");
    client.println("Host: 192.168.1.2");
    client.println("Connection: close");
    client.println();
    // Очікування відповіді від сервера
    while (client.connected()) {
      if (client.available()) {
        String line = client.readStringUntil('\n');
        if (line == "\r") {
          break;
    // Отримання команди від сервера
    String command = client.readStringUntil('\n');
    Serial.println("Received command: " + command);
    if (command == "ACTIVATE_SIREN") {
      digitalWrite(sirenPin, HIGH);  // Активувати сирену
    } else if (command == "DEACTIVATE_SIREN") {
      digitalWrite(sirenPin, LOW);
    } else if (command == "ACTIVATE_LIGHT") {
      digitalWrite(lightPin, HIGH);  // Активувати світловий сигнал
```

```
    } else if (command == "DEACTIVATE_LIGHT") {
      digitalWrite(lightPin, LOW);   // Деактивувати світловий сигнал
    }
    client.stop();
    return true;
  } else {
    return false;
void initializeGSM() {
  gsmSerial.println("AT");
  delay(1000);
  gsmSerial.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"");
  delay(1000);
  gsmSerial.println("AT+SAPBR=3,1,\"APN\",\"" + String(apn) + "\"");
  delay(1000);
  gsmSerial.println("AT+SAPBR=1,1");
  delay(1000);
  gsmSerial.println("AT+SAPBR=2,1");
  delay(1000);
}
```

Модуль резервних каналів даних є критичним компонентом системи оповіщення про ракетну небезпеку, що забезпечує надійний та безперервний зв'язок між апаратною та серверною частинами. Використання альтернативних засобів зв'язку, таких як GSM, дозволяє підтримувати роботу системи навіть у випадку виходу з ладу основного каналу. Це забезпечує високу надійність та ефективність системи оповіщення, що є критично важливим для своєчасного попередження населення про загрозу.

### 3.4 Аналіз та тестування реалізованих засобів

Після завершення розробки та реалізації системи оповіщення про ракетну небезпеку необхідно провести детальний аналіз та тестування її компонентів для перевірки функціональності, надійності та ефективності. Цей процес включає кілька етапів: модульне тестування, інтеграційне тестування, системне тестування та тестування в реальних умовах.

Модульне тестування полягає в перевірці окремих компонентів системи на правильність виконання функцій, передбачених технічними вимогами. Кожен модуль тестується ізольовано від інших для виявлення та виправлення можливих помилок.

Протестуємо роботу серверної частини (рис. 3.1).

```
[Maliths-MBP:rlog malith$
[Maliths-MBP:rlog malith$
[Maliths-MBP:rlog malith$ make
 Building object file: main.o
 g++ -o build/main.o -c src/main.cc -std=c++11

 Linking object file main with main.o
 g++ -o build/main build/main.o -std=c++11 -I include

[Maliths-MBP:rlog malith$
[Maliths-MBP:rlog malith$ ./build/main
 05-04-2020 10:35:31 INFO: src/main.cc initializing the logging process
 05-04-2020 10:35:31 INFO: src/main.cc Hello!
 05-04-2020 10:35:31 INFO: src/buffer.h initializing the logging process
 05-04-2020 10:35:31 INFO: src/buffer.h Aloha!
 05-04-2020 10:35:31 WARN: src/buffer.h temporarily snoozing down the system
 05-04-2020 10:35:33 ERROR: src/buffer.h critical computation time lost
 05-04-2020 10:35:33 INFO: src/buffer.h retrying to start the system: attempt: 0
 05-04-2020 10:35:33 WARN: src/buffer.h temporarily snoozing down the system
 05-04-2020 10:35:35 ERROR: src/buffer.h critical computation time lost
 05-04-2020 10:35:35 INFO: src/buffer.h retrying to start the system: attempt: 1
 05-04-2020 10:35:35 WARN: src/buffer.h temporarily snoozing down the system
 05-04-2020 10:35:37 ERROR: src/buffer.h critical computation time lost
 05-04-2020 10:35:37 INFO: src/buffer.h retrying to start the system: attempt: 2
 05-04-2020 10:35:37 WARN: src/buffer.h temporarily snoozing down the system
 05-04-2020 10:35:39 ERROR: src/buffer.h critical computation time lost
 05-04-2020 10:35:39 INFO: src/buffer.h retrying to start the system: attempt: 3
 05-04-2020 10:35:39 WARN: src/buffer.h temporarily snoozing down the system
 05-04-2020 10:35:41 ERROR: src/buffer.h critical computation time lost
 05-04-2020 10:35:41 INFO: src/buffer.h retrying to start the system: attempt: 4
 05-04-2020 10:35:41 INFO: src/buffer.h resumed all systems
 05-04-2020 10:35:41 INFO: src/buffer.h allocating memory on the buffer. requeste
 05-04-2020 10:35:41 INFO: src/buffer.h tensor requested for space from the buffe
 05-04-2020 10:35:41 WARN: src/buffer.h space not available on the buffer. expand
 05-04-2020 10:35:41 INFO: src/buffer.h tensor was granted with the required spac
```

Рисунок 3.1 – Логи роботи сервера

Судячи з логів сервера можнра зробити висновок, що він запускається та працює корекно.

Перевіримо роботу апараної чатини (рисю 3.2).

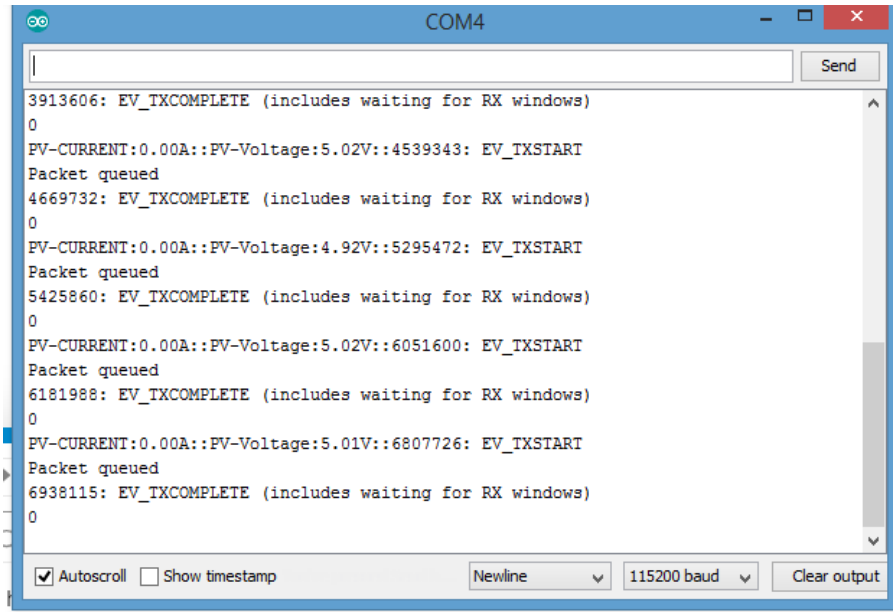



Рисунок 3.2 – Логи роботи пристрою

Судячи з логів можна впевнено сказати, що апратна частина праццює справно.

Інтеграційне тестування полягає в перевірці взаємодії між окремими модулями системи для забезпечення їх коректної роботи в складі єдиної системи. На цьому етапі виявляються проблеми, пов'язані з інтеграцією модулів.

Перевіримо взаємлдію серверної та апратної частини надіславши запит про тривогу від сервера до пристрою (рис.3.3).

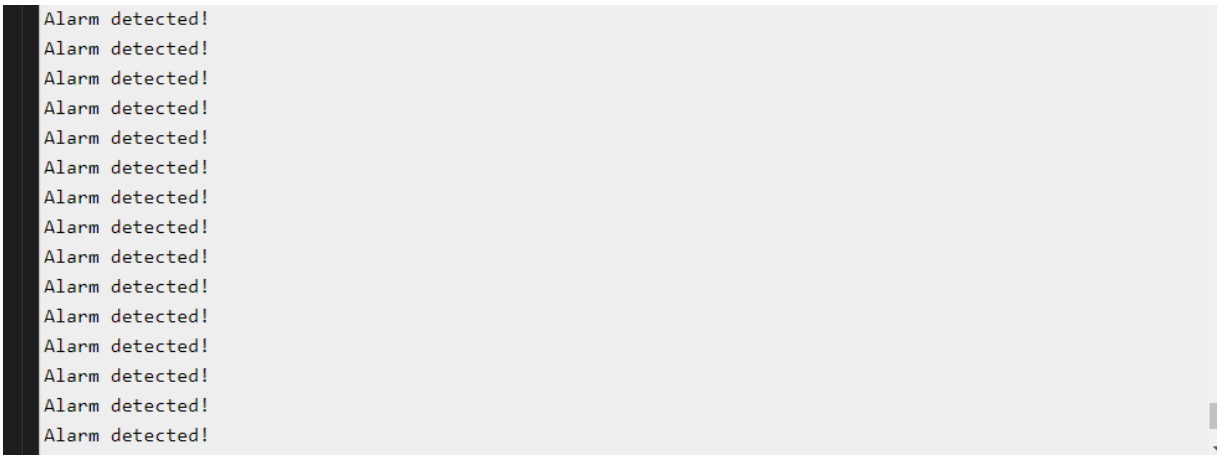



Рисунок 3.3 – Вигляд консолі пристрою при отримані сигналу про тривогу

Як можна побачити на рисунку 3.3, пристрій коректно взаємодіє разом з серверною частиною.

Системне тестування включає перевірку всієї системи як цілісного комплексу, забезпечуючи її коректну роботу у всіх передбачених режимах та умовах. Цей етап передбачає тестування всіх функцій системи в автоматичному та ручному режимах, а також перевірку стійкості системи до різних збоїв.

Автоматичний режим роботи системи передбачає автономне функціонування без втручання оператора. Основною метою цього етапу тестування є перевірка коректності роботи системи у разі виявлення загрози.

Первевіримо правильність обробки тривоги на всіх етапах, спочатку перевіримо чи серверна частина оброблює сигнал про тривогу та надсилає його апаратним пристроям (рис.3.4).

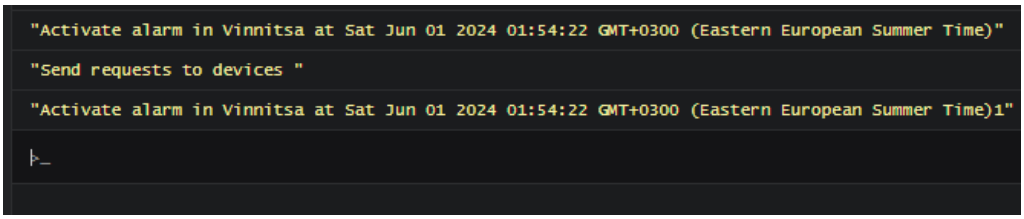



Рисунок 3.4 – Логи про отримання сигналу про повітряну тривогу на серверній частині

Як видно серверна частина правильно оролює сигнали про повітряну тривогу.

Наступним короком перевіримо чи вірно оролює сигнали про тривогу апратна чатсина (рис. 3.5).


Рисунок 3.5 – Отриманням повідомлення про тривогу від серверної частини

Отже, можна зробити висновок, що взаємодія серверної та апаратної частини в автоматичному режимі є налаштованою та працеспроможною.

Ручний режим роботи системи передбачає можливість керування системою оператором. Основною метою цього етапу тестування є перевірка коректності обробки команд, що надходять від оператора.

Протестуємо роботу системи в ручному режимі, для цього відішлемо ручне повідомлення про тривогу (рис.3.6).

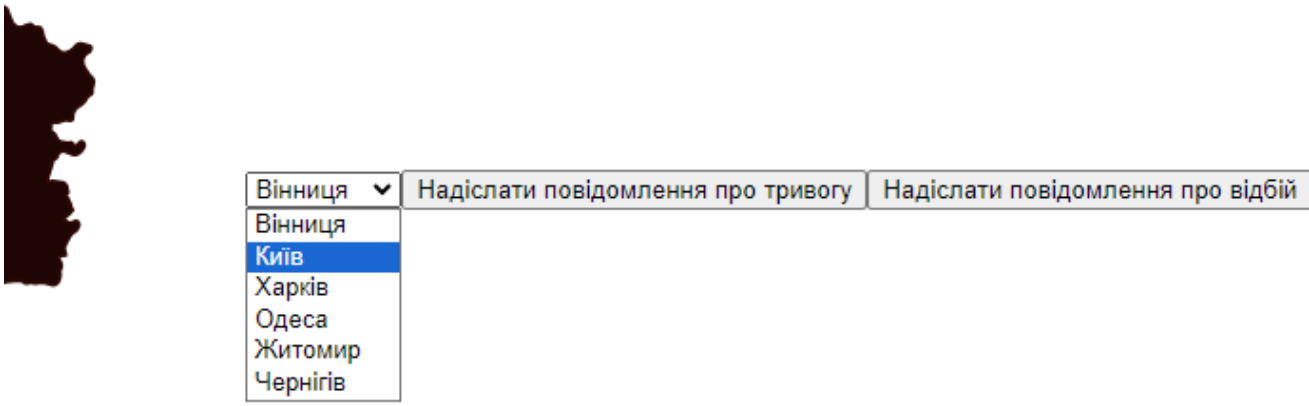



Рисунок 3.6 – Надсилання повідомлення про тривогу в ручному режимі

Перевіримо чи надійшов сигнал до апаратного пристрою (рис. 3.7).

```
Manual alarm detected!
Detect manual request to alarm
Activate manual  alarm in Vinnitsa
Manual alarm detected!
```

Рисунок 3.7 – Отримання повідомлення про ручне включення системи оповіщення

Як можна побачити на рисунку 3.7 є логи про отримання повідомлення про ручну активацію сповіщення про повітряну тривогу, що дає змогу зробити висновок, що ручна активація повідолмення прпо трривогу працює коректно.

Наступним кроком протестуємо інтерфес відображення карти повітряних тривог, який дозволяє зручно спостерігати за ситуацією в країні (рис. 3.8).

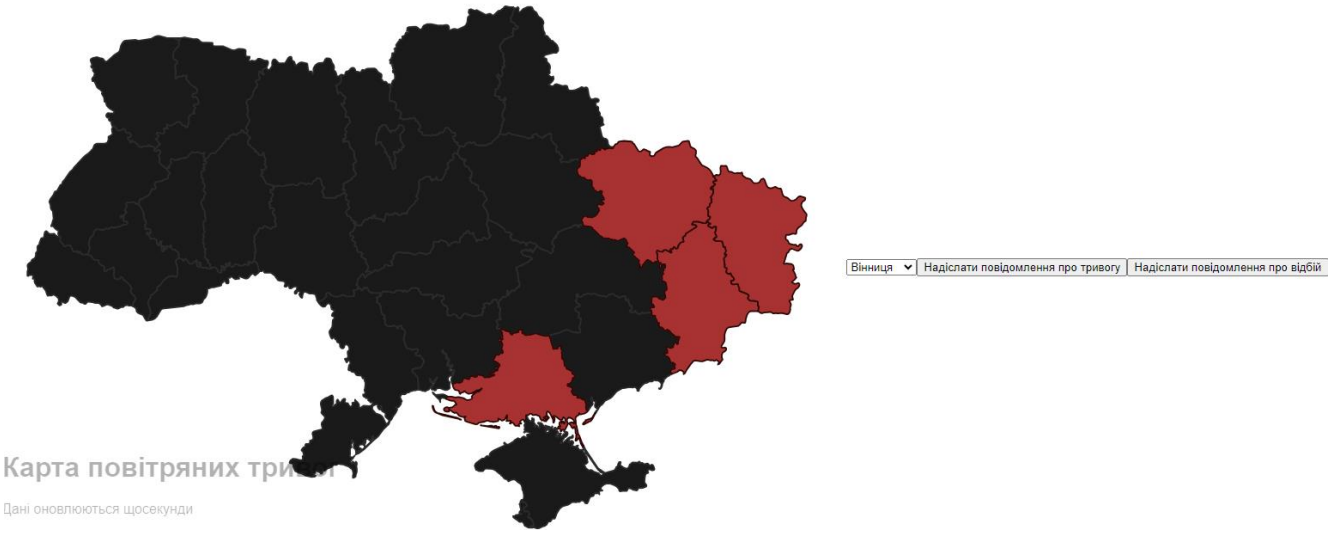



Рисунок 3.8 – Карта відображення тривог

Як видно відображжження карти тривог працює вірно.

Наступним кроком протестуємо процес аунтентифікації користувача (рис. 3.9).

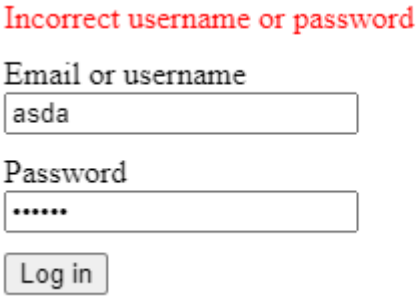



Рисунок 3.9 - Спроба аунтентифікації в додатку з невірними даними

Як видно з рисунку 3.9, користувач не може авторизуватися з невірними даними, що свідчить про правильність роботи додатку.

Завдяки ретельному тестуванню всіх компонентів системи вдалося досягти високої надійності та ефективності її роботи. Система оповіщення про ракетну небезпеку продемонструвала свою здатність своєчасно та надійно інформувати населення про загрозу, використовуючи як основні, так і резервні канали зв’язку, забезпечуючи безпеку переданих даних. Це підтверджує готовність системи до впровадження та використання у реальних умовах для захисту населення від ракетних загроз.

### 3.5 Висновки до розділу

У цьому розділі була проведена комплексна розробка та тестування програмних і апаратних компонентів захищеної системи оповіщення про ракетну небезпеку, яка включає автоматичний та ручний режими роботи, односторонній зв'язок без зворотного зв'язку, резервні канали передачі даних та шифрування інформації. Основні висновки за результатами виконаної роботи наведені нижче.

Вибір мови програмування C++ для розробки серверної частини та апаратної частини на Arduino був обґрунтований високою продуктивністю, гнучкістю та можливістю ефективного управління ресурсами. Використання Visual Studio як середовища розробки забезпечило зручність розробки, налагодження та тестування програмного забезпечення.

Серверна частина системи була розроблена з використанням C++ та Visual Studio. Основні модулі включали модуль обробки даних, модуль захисту даних, модуль управління базою даних та модуль резервних каналів даних. Кожен модуль був реалізований відповідно до визначених вимог та забезпечував надійну та ефективну роботу системи.

Проведено детальне модульне, інтеграційне, системне тестування та тестування в реальних умовах. Система продемонструвала високу функціональну коректність, інтеграційну сумісність, стійкість до збоїв та здатність ефективно функціонувати в реальних умовах експлуатації. Захист даних був забезпечений на високому рівні завдяки реалізованим алгоритмам шифрування та дешифрування.

Розробка та тестування захищеної системи оповіщення про ракетну небезпеку показали, що обрані технічні рішення та засоби відповідають поставленим вимогам. Система забезпечує своєчасне та надійне оповіщення населення про ракетні загрози, використовуючи як основні, так і резервні канали зв’язку, а також надає високий рівень безпеки переданих даних. Впровадження цієї системи сприятиме підвищенню рівня безпеки та захисту населення у випадках ракетних загроз.

# 4 ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає сучасним вимогам як у напрямку науково-технічного прогресу, так і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність отриманих результатів.

Магістерська кваліфікаційна робота за темою «Захищена система оповіщення населення про ракетну небезпеку в автоматичному та “ручному” режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних» належить до науково-технічних робіт, орієнтованих на виведення на ринок. Рішення про комерціалізацію цієї розробки може бути прийняте в процесі проведення самої роботи. Такий напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект.

Для успішної реалізації проекту необхідно знайти потенційного інвестора, який би взявся за впровадження цього проекту, і переконати його в економічній доцільності такого кроку. Підтвердженням цього може бути прогнозована вигода, швидкий термін окупності та високий комерційний потенціал розробки.

## 4.1 Оцінювання комерційного потенціалу розробки програмного забезпечення

Метою комерційного і технологічного аудиту є оцінка науково-технічного рівня та комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності, зокрема під час виконання магістерської кваліфікаційної роботи. Для проведення такого аудиту залучають не менше трьох незалежних експертів, серед яких можуть бути провідні викладачі випускової або спорідненої кафедри, а також інші відомі фахівці [40].

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п’ятибальної системи оцінювання за 12-ма критеріями, наведеними в таблиці 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

| Бали (за 5-ти бальною шкалою) | | | | | |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 |
| Технічна здійсненність концепції | | | | | |
| 1 | Достовірність концепції не підтверджена | Концепція підтверджена експертними висновками | Концепція підтверджена розрахунками | Концепція перевірена на практиці | Перевірено працездатність продукту в реальних умовах |
| Ринкові переваги (недоліки) | | | | | |
| 2 | Багато аналогів на малому ринку | Мало аналогів на малому ринку | Кілька аналогів на великому ринку | Один аналог на великому ринку | Продукт не має аналогів на великому ринку |
| 3 | Ціна продукту значно вища за ціни аналогів | Ціна продукту дещо вища за ціни аналогів | Ціна продукту приблизно дорівнює цінам аналогів | Ціна продукту дещо нижче за ціни аналогів | Ціна продукту значно нижче за ціни аналогів |
| 4 | Технічні та споживчі властивості продукту значно гірші, ніж в | Технічні та споживчі влас-тивості продукту трохи гірші, ніж в аналогів | Технічні та споживчі властивості продукту на рівні аналогів | Технічні та споживчі властивості продукту трохи кращі, ніж в | Технічні та споживчі властивості продукту значно кращі, ніж в |
| 5 | Експлуатаційні витрати значно вищі, ніж в аналогів | Експлуатаційні витрати дещо вищі, ніж в аналогів | Експлуатаційні витрати на рівні експлуатаційних витрат аналогів | Експлуатаційні витрати трохи нижчі, ніж в аналогів | Експлуатаційні витрати значно нижчі, ніж в аналогів |
| Ринкові перспективи | | | | | |
| 6 | Ринок малий і не має позитивної динаміки | Ринок малий, але має позитивну динаміку | Середній ринок з позитивною динамікою | Великий стабільний ринок | Великий ринок з позитивною динамікою |
| 7 | Активна конкуренція великих компаній на | Активна конкуренція | Помірна конкуренція | Незначна конкуренція | Конкурентів немає |
| Практична здійсненність | | | | | |
| 8 | Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї | Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців | Необхідне незначне навчання фахівців та збільшення їх штату | Необхідне незначне навчання фахівців | Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї |

Продовження таблиці 4.1

| | | | | | |
|---|---|---|---|---|---|
| 9 | Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні | Потрібні незначні фінансові ресурси. Джерела фінансування відсутні | Потрібні значні фінансові ресурси. Джерела фінансування є | Потрібні незначні фінансові ресурси. Джерела фінансування є | Не потребує додаткового фінансування |
| 10 | Необхідна розробка нових матеріалів | Потрібні матеріали, що використовуються у військово промисловому комплексі | Потрібні дорогі матеріали | Потрібні досяжні та дешеві матеріали | Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві |
| 11 | Термін реалізації ідеї більший за 10 років | Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років | Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років | Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років | Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років |
| 12 | Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту | Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу | Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу | Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту | Відсутні будь- які регламентні обмеження на виробництво та реалізацію продукту |

Результати оцінки науково-технічного рівня та комерційного потенціалу науково-технічної розробки слід представити у вигляді таблиці.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

| Критерії | Прізвище, ініціали, посада експерта | | |
|---|---|---|---|
| | 1. | 2. | 3. |
| 1 | 4 | 4 | 5 |
| Ринкові переваги (недоліки): | | | |
| 2 | 3 | 3 | 4 |
| 3 | 5 | 4 | 3 |

Продовження таблиці 4.2

| | | | |
|---|---|---|---|
| 4 | 4 | 5 | 4 |
| 5 | 4 | 3 | 5 |
| Ринкові перспективи | | | |
| 6 | 4 | 4 | 4 |
| 7 | 3 | 4 | 3 |
| Практична здійсненність | | | |
| 8 | 3 | 4 | 4 |
| 9 | 3 | 4 | 5 |
| 10 | 5 | 5 | 4 |
| 11 | 4 | 3 | 4 |
| 12 | 5 | 4 | 3 |
| Сума балів | $СБ_1 = 47$ | $СБ_1 = 47$ | $СБ_1 = 51$ |
| Середньоарифметична сума балів $СБ_с$ | $СБ = 48,3$ | | |

На основі даних, що представлені у таблиці 4.2, можна зробити висновок про рівень комерційного потенціалу розробки. Порівняємо ці результати з рівнями комерційного потенціалу, які наведені у таблиці 4.3.

Таблиця 4.3 – Рівні комерційного потенціалу розробки

| Середньоарифметична сума балів СБ, розрахована на основі висновків експертів | Рівень комерційного потенціалу розробки |
|---|---|
| 0 – 10 | Низький |
| 11 – 20 | Нижче середнього |
| 21 – 30 | Середній |
| 31 – 40 | Вище середнього |
| 41 – 48 | Високий |

За результатами наших досліджень було встановлено, що розробка системи захищеного оповіщення населення про ракетну небезпеку, яка працює як у автоматичному, так і "ручному" режимах, з використанням одностороннього зв'язку, резервних каналів обміну та шифрування даних, має рівень комерційного потенціалу на рівні 48,3 балів. Згідно з таблицею 4.3, це свідчить про високу комерційну важливість проведення цих досліджень.

## 4.2 Прогнозування витрат на виконання наукової роботи та впровадження її результатів

Прогнозування витрат на проведення науково-дослідної, дослідно-конструкторської та конструкторсько-технічної роботи включає ретельний аналіз різних аспектів витрат та вплив на виконання проекту.

В першу чергу, визначаються витрати, які прямо пов'язані зі зусиллями та ресурсами, витраченими на виконання цієї роботи. Сюди включаються витрати на оплату праці, навчання та інші витрати, які безпосередньо пов'язані з реалізацією проекту.

Другий аспект - це розрахунок загальних витрат на виконання всієї роботи. Це охоплює витрати на матеріали, обладнання, послуги та інші загальні витрати, пов'язані із проектом у цілому.

Останній аспект полягає в прогнозуванні загальних витрат на впровадження результатів роботи. Це включає витрати на впровадження розробок, рекламу, підготовку персоналу та інші витрати, пов'язані з реалізацією отриманих результатів [40].

Ці аспекти витрат необхідно ретельно розглядати для забезпечення успішного виконання проекту.

До категорії "Витрати на оплату праці" включаються витрати, пов'язані з виплатою основної та додаткової заробітної плати працівникам, що обіймають посади керівників відділів, лабораторій, секторів та груп, науковим та інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, які безпосередньо зайняті виконанням конкретної теми [40].

Ці витрати розраховуються за посадовими окладами, відрядними розцінками, тарифними ставками відповідно до систем оплати праці, що діють у відповідних організаціях. Крім того, до цієї категорії відносяться будь-які види грошових і матеріальних надбавок, які включаються до складу "Витрат на оплату праці".

Витрати на основну заробітну плату дослідників ($З_о$) розраховують відповідно до посадових окладів працівників, за формулою:

Основна заробітна плата $З_о$:

$$З_о = \sum_{i=1}^{k} \frac{M_{ni} \cdot t_i}{T_p} \quad (4.1)$$

де $k$ – кількість посад дослідників залучених до процесу досліджень;

$M_{ni}$ – місячний посадовий оклад конкретного дослідника, грн;

$t_i$ – число днів роботи конкретного дослідника, дні;

$T_p$ – середнє число робочих днів в місяці, $T_p$= 21 дня.

$$З_о = \frac{37600}{21} * 40 = 62\,616$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.4 – Витрати на заробітну плату дослідників

| Найменування посади | Місячний посадовий оклад, грн | Оплата за робочий день, грн | Число днів роботи | Витрати на заробітну плату, грн |
|---|---|---|---|---|
| Координатор проекту | 37600 | 1790,5 | 43 | 76990,5 |
| Інженер з розробки програмного забезпечення | 32400 | 1542,9 | 40 | 61714,3 |
| Тестувальник програмного забезпечення | 30200 | 1438,1 | 15 | 21571,4 |
| | | | Всього | 160276,2 |

Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт НДР розраховуємо за формулою:

$$З_р = \sum_{i=1}^{n} C_i \cdot t_i \quad (4.2)$$

де $C_i$ – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

$t_i$ – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду $C_i$ можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (4.3)$$

де $M_M$ – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), приймемо *Мм* = 6800,00 грн;

$K_i$ – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду.

$K_c$ – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

$T_p$ – середнє число робочих днів в місяці, приблизно $T_p$ = 21 день;

$t_{зм}$ – тривалість зміни, год.

$$\mathrm{C_1} = \frac{8000 \ * 1.10 \ * 1{,}65}{21 \ * 8} = 82{,}5 \text{ грн.}$$

$$\mathrm{Зр_1} = 82{,}5 \ * 4 = 330 \text{ грн.}$$

Таблиця 4.5 – Величина витрат на основну заробітну плату робітників

| Найменування робіт | Тривалість роботи, год | Розряд роботи | Тарифний коефіцієнт | Погодинна тарифна ставка, грн | Величина оплати на робітника грн |
|---|---|---|---|---|---|
| Розміщення комп'ютерного обладнання | 4 | 2 | 1,1 | 82,5 | 330 |
| Адміністрування системи | 3 | 2 | 1,1 | 82,5 | 247,5 |
| Розгортання програмного забезпечення | 3 | 5 | 1,7 | 127,5 | 385,5 |
| | | | | Всього | 963 |

Додаткову заробітну плату розраховуємо як 10…12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = (З_о + З_р) \cdot \frac{Н_{дод}}{100\%} \quad (4.4)$$

де $Н_{дод}$ – норма нарахування додаткової заробітної плати. $Н_{дод}$ - приймемо, як 12%.

$$З_{дод} = (160276{,}2 + 963) * \frac{12}{100} = 19348{,}7 \text{ грн.}$$

До статті "Відрахування на соціальні заходи" включаються внески на загальнообов'язкове державне соціальне страхування та витрати на соціальний захист населення, зокрема єдиний соціальний внесок (ЄСВ) [40].

Нарахування на заробітну плату дослідників та працівників становить 22% від суми їх основної та додаткової заробітної плати і розраховується за наступною формулою:

$$З_н = (З_о + З_р + З_{дод}) \cdot \frac{Н_{зп}}{100\%} \quad (4.5)$$

де $Н_{зп}$ – норма нарахування на заробітну плату.

$$З_н = (160276{,}2 + 963 + 19348{,}7) * \frac{22}{100} = 39729{,}3 \text{ грн.}$$

До статті «Сировина та матеріали» відносяться витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, придбані у сторонніх підприємств, установ і організацій та використані для проведення досліджень за прямим призначенням згідно з нормами їх витрачання. Також до цієї статті включаються витрати на придбані напівфабрикати, що потребують монтажу, виготовлення або додаткової обробки в даній організації, а також дослідні зразки, виготовлені виробниками за документацією наукової організації [41].

Вартість матеріалів (М) розраховується окремо для кожного виду матеріалів за наступною формулою:

$$M = \sum_{j=1}^{n} H_j \cdot Ц_j \cdot К_j - \sum_{j=1}^{n} B_j \cdot Ц_{вj}, \quad (4.6)$$

де $H_j$ – норма витрат матеріалу $j$-го найменування, кг;

$n$ – кількість видів матеріалів;

$Ц_j$ – вартість матеріалу $j$-го найменування, грн/кг;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = *1,1*);

$B_j$ – маса відходів $j$-го найменування, кг;

$Ц_{вj}$ – вартість відходів $j$-го найменування, грн/кг.

$$М_1 = 234 * 2 * 1{,}1 - 0{,}0 - 0{,}0 = 514{,}8 \text{ грн.}$$

Таблиця 4.6 – Витрати на матеріали

| Найменування матеріалу, марка, тип, сорт | Ціна за од, грн | Норма витрат, од | Величина відходів, кг | Ціна відходів, грн/кг | Вартість витраченого матеріалу, грн |
|---|---|---|---|---|---|
| Папір формату А4 | 1 | 234,00 | 0 | 0 | 514,8 |
| Папка для документів | 2 | 54,00 | 0 | 0 | 118,8 |
| Самоклеючі записки | 2 | 27,00 | 0 | 0 | 59,4 |
| Набір для письма | 2 | 185,20 | 0 | 0 | 407,44 |
| Файли для документів | 1 | 81,00 | 0 | 0 | 89,1 |
| Серветки для чищення техніки | 1 | 159,00 | 0 | 0 | 174,9 |
| | | | | Всього | 1364,44 |

Витрати на комплектуючі вироби ($К_в$), які використовують при дослідженні нового технічного рішення, розраховуються, згідно з їхньою номенклатурою, за формулою:

$$К_в = \sum_{j=1}^{n} H_j \cdot Ц_j \cdot К_j \quad (4.7)$$

де $Н_j$ – кількість комплектуючих j-го виду, шт.;

$Ц_j$ – покупна ціна комплектуючих j-го виду, грн;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = 1,1 … 1,15).

$$К_в = 1 * 300 * 1{,}1 = 330{,}00 \text{ грн.}$$

| Найменування комплектуючих | Кількість, шт. | Ціна за штуку, грн | Сума, грн |
|---|---|---|---|
| NodeMCU | 1 | 300,00 | 330,00 |
| GCM module | 1 | 120,00 | 132,00 |
| Ethernet module | 1 | 80,00 | 88,00 |
| Акумуляторна батарея 18650 | 1 | 180,00 | 198,00 |
| Провідні матеріали | 1 | 20,00 | 22,00 |
| Кейс для акумулятора | 1 | 20,00 | 22,00 |
| | | Всього | 792,00 |

До статті "Спецустаткування для наукових (експериментальних) робіт" включаються витрати на виготовлення та придбання спеціального устаткування, необхідного для проведення досліджень, а також витрати на його проектування, виготовлення, транспортування, монтаж та встановлення. У даній дослідницькій роботі витрати на спецустаткування відсутні [41].

До статті «Програмне забезпечення для наукових (експериментальних) робіт» включаються витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення (програм, алгоритмів, баз даних), необхідних для проведення досліджень, а також витрати на їх проектування, створення та встановлення.

Балансова вартість програмного забезпечення розраховується за формулою:

$$B_{прг} = \sum_{i=1}^{k} Ц_{iпрг} \cdot C_{прг.i} \cdot K_i \ , \qquad (4.8)$$

де $Ц_{iпрг}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{прг.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$ – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i$ = 1,10);

$k$ – кількість найменувань програмних засобів.

$$\text{В}_{\text{прг}} = 7\,200 \times 1 \times 1{,}1 = 7920 \text{ грн.}$$

Таблиця 4.7 – Витрати на придбання програмних засобів по кожному виду

| Найменування програмного засобу | Кількість, шт | Ціна за одиницю, грн | Вартість, грн |
|---|---|---|---|
| dotUltimate | 1 | 7200,00 | 7920,00 |
| CLion | 1 | 8450,00 | 9295,00 |
| DataSpell | 1 | 3980,00 | 4378,00 |
| | | Всього | 21593,00 |

До статті «Амортизація обладнання, програмних засобів та приміщень» включаються амортизаційні відрахування за кожним видом обладнання, устаткування, інших приладів і пристроїв, а також програмного забезпечення, які використовуються для проведення науково-дослідної роботи, за їх наявності в дослідницькій організації або на підприємстві.

У спрощеному вигляді амортизаційні відрахування за кожним видом обладнання, приміщень та програмного забезпечення можуть бути розраховані за допомогою прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_в} \cdot \frac{t_{вик}}{12}, \quad (4.9)$$

де $Ц_б$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$\text{А}_{\text{обл}} = \frac{35\,000 \times 2}{3 \times 12} = 1944{,}4 \text{ грн.}$$

Таблиця 4.8 – Амортизаційні відрахування по кожному виду обладнання

| Найменування обладнання | Балансова вартість, грн | Строк корисного використання, років | Термін використання обладнання, місяців | Амортизаційні відрахування, грн |
|---|---|---|---|---|
| Ноутбук Dream Machines RG3050-15 | 35000,00 | 3 | 2 | 1944,4 |
| Ноутбук ASUS TUF Gaming FA506NF-HN033 | 32500,00 | 3 | 2 | 1805,5 |
| Робоче місце розробника | 16800,00 | 5 | 2 | 560,00 |
| Принтер Epson Eco Tank L1250 | 7225,00 | 4 | 2 | 301,00 |
| | | | Всього | 4610,9 |

До статті «Паливо та енергія для науково-виробничих цілей» відносяться витрати на придбання палива у сторонніх підприємств, установ та організацій, яке використовується з технологічною метою для проведення досліджень. Ця стаття формується у разі проведення енергоємних наукових досліджень за методом прямого віднесення витрат і може становити значну частку у собівартості досліджень.

Витрати на силову електроенергію ($В_е$) розраховуються за формулою:

$$B_e = \sum_{i=1}^{n} \frac{W_{yi} \cdot t_i \cdot Ц_e \cdot K_{вni}}{\eta_i}, \quad (4.10)$$

де $W_{yi}$ – встановлена потужність обладнання на визначеному етапі розробки, кВт;

$t_i$ – тривалість роботи обладнання на етапі дослідження, год;

$Ц_e$ – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії).

Приймемо $Ц_е$ = 7,50 грн;

$K_{вп і}$ – коефіцієнт, що враховує використання потужності, $K_{вп і} < 1$;

$\eta_i$ – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$$В_е = \frac{0{,}46 \times 320 \times 7{,}50 \times 0{,}95}{0{,}97} = 837{,}4 \text{ грн.}$$

Таблиця 4.9 – Витрати на електроенергію

| Найменування обладнання | Встановлена потужність, кВт | Тривалість роботи, год | Сума, грн |
|---|---|---|---|
| Ноутбук Dream Machines RG3050-15 | 0,46 | 320 | 1081,2 |
| Ноутбук ASUS TUF Gaming FA506NF-HN033 | 0,48 | 352 | 1241,0 |
| Робоче місце розробника | 0,25 | 355 | 651,9 |
| Принтер Epson Eco Tank L1250 | 0,12 | 18 | 15,9 |
| | | Всього | 2990,00 |

До статті «Службові відрядження» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень [41].

Витрати за статтею «Службові відрядження» розраховуються як 20…25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{св} = (З_о + З_р) \cdot \frac{H_{св}}{100\%}, \quad (4.11)$$

де $H_{св}$ – норма нарахування за статтею «Службові відрядження», приймемо $Н_{св} = 20\%$.

$$В_{св} = (160276{,}2 + 963{,}00) \times \frac{20}{100} = 32247{,}84 \text{ грн.}$$

До статті «Витрати на роботи, які виконують сторонні підприємства, установи і організації» належать витрати на проведення досліджень, що не можуть бути виконані штатними працівниками або наявним обладнанням організації, а

виконуються на договірній основі іншими підприємствами, установами і організаціями незалежно від форм власності та позаштатними працівниками.

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» розраховуються як 30…45% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{сп} = (З_о + З_р) \cdot \frac{H_{сп}}{100\%}, \quad (4.12)$$

де $Н_{сп}$ – норма нарахування за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації», приймемо $Н_{сп}$= 40%.

$$В_{сп} = (160276{,}2 + 963{,}00) \times \frac{40}{100} = 64495{,}68 \text{ грн.}$$

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуються як 50…100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_в = (З_о + З_р) \cdot \frac{H_{ів}}{100\%}, \quad (4.13)$$

де $H_{ів}$ – норма нарахування за статтею «Інші витрати», приймемо $Н_{ів}$ = 60%.

$$І_в = (160276{,}2 + 963{,}00) \times \frac{60}{100} = 96743{,}5 \text{ грн.}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов’язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов’язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов’язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін [41].

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100…150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = (З_о + З_р) \cdot \frac{Н_{нзв}}{100\%}\ , \qquad (4.14)$$

де $Н_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», приймемо $Н_{нзв}$ = 100%.

$$В_{нзв} = (160276{,}2 + 963{,}00) \times \frac{100}{100} = 161239{,}2 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат за формулою:

$$B_{заг} = З_о + З_р + З_{дод} + З_н + М + К_в + В_{спец} + В_{прг} + А_{обл} + В_е + В_{св} + В_{сп} + І_в + В_{нзв} \qquad (4.15)$$

$$В_{заг} = 160276{,}2 + 963{,}00 + 19348{,}7 + 39729{,}3 + 1364{,}44 + 792{,}00 + 21593{,}00 + 4610{,}9 + 2990{,}00 + 32247{,}84 + 64495{,}68 + 96743{,}5 + 161239{,}2 = 606\ 393{,}76 \text{ грн.}$$

Загальні витрати ЗВ на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \qquad (4.16)$$

де $\eta$ - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, приймемо $\eta$=0,7.

$$ЗВ = \frac{606\ 393{,}76}{0{,}7} = 866276{,}8 \text{ грн.}$$

## 4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

У ринкових умовах позитивний результат від можливого впровадження науково-технічної розробки для потенційного інвестора полягає у збільшенні чистого прибутку. Дослідження щодо "Захищеної системи оповіщення населення про ракетну небезпеку в автоматичному та «ручному» режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних" передбачають комерціалізацію протягом трьох років.

У цьому контексті майбутній економічний ефект базується на зростанні кількості користувачів продукту протягом аналізованого періоду:

– перший рік – 5000 користувачів;

– другий рік – 6500 користувачів;

– третій рік – 7000 користувачів.

Кількість користувачів, які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки, становила $N$=10000 осіб.

Вартість програмного продукту у році до впровадження результатів розробки становила $Ц_б$=18000,00 грн.

Зміна вартості програмного продукту від впровадження результатів науково-технічної розробки приймається на рівні $\pm\Delta Ц_о$=5000,00 грн.

Для кожного з випадків потенційне збільшення чистого прибутку у потенційного інвестора (ΔПі) в роки очікуваного позитивного результату від можливого впровадження та комерціалізації науково-технічної розробки розраховується за відповідною формулою:

$$\Delta П_i = (\pm\Delta Ц_о \cdot N + Ц_о \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\vartheta}{100}), \quad (4.17)$$

де $\lambda$ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2024 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda$ *=0,8333*;

$\rho$ – коефіцієнт, який враховує рентабельність інноваційного продукту. Приймемо $\rho$=30*%*;

$\vartheta$ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2024 році $\vartheta$ *=18%*;

$$\Delta П_1 = (5000 * 10000 + 18000 * 5000) * 0{,}83 * 0{,}3 * \left(1 - \frac{0{,}18}{100}\right)$$

$$= 34\ 797\ 252 \text{ грн.}$$

$$\Delta П_2 = \left(5000 * 10000 + 18000 * (5000 + 6500)\right) * 0{,}83 * 0{,}3 * \left(1 - \frac{0{,}18}{100}\right)$$

$$= 63\ 877\ 812{,}6 \text{ грн.}$$

$$\Delta П_3 = \left(5000 * 10000 + 18000 * (5000 + 6500 + 7000)\right) * 0{,}83 * 0{,}3 * \left(1 - \frac{0{,}18}{100}\right)$$

$$= 95\ 195\ 339{,}4 \text{ грн.}$$

Далі розраховують приведену вартість збільшення всіх чистих прибутків ПП, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \tag{4.18}$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

$T$ – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

$\tau$ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau$=0,2;

$t$ – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$\text{ПП} = \frac{34\ 797\ 252}{(1+0{,}2)^1} + \frac{63\ 877\ 812{,}6}{(1+0{,}2)^2} + \frac{95\ 195\ 339{,}4}{(1+0{,}2)^3} = 128\ 447\ 197{,}5 \text{ грн.}$$

Отже, згідно з розрахунками, впровадження цієї розробки принесе значну комерційну вигоду, що підтверджує прогнози та призведе до зростання чистого прибутку підприємства.

## 4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{інв} \cdot ЗВ, \tag{4.19}$$

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{інв}$=3;

$ЗВ$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 866276,8 грн.

$$\text{PV} = 3 \times 866276{,}8 = 2585852{,}4 \text{ грн.}$$

Тоді абсолютний економічний ефект $\text{Е}_{\text{абс}}$ або чистий приведений дохід для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV \tag{4.20}$$

де $ПП$ – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 128 447 197,5 грн;

$PV$ – теперішня вартість початкових інвестицій, 2598830,4 грн.

$$\text{Е}_{\text{абс}} = 128\,447\,197{,}5 - 2598830{,}4 = 125\,848\,367{,}1 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій $\text{Е}_{\text{в}}$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки, розраховується за формулою:

$$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1, \tag{4.21}$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, 125 848 367,1 грн;

$PV$ – теперішня вартість початкових інвестицій, 2598830,4 грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 3 роки.

$$\text{Е}_{\text{в}} = \sqrt[3]{1 + \frac{125\,848\,367{,}1}{2598830{,}4}} - 1 = 2{,}6$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій мін τ визначається за формулою:

$$\tau_{мін} = d + f\,, \tag{4.22}$$

де $d$ – середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні $d$ =0,4;

$f$ – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,2.

$$\tau_{min} = 0{,}2 + 0{,}4 = 0{,}6$$

$$E_в = 260\% > \tau_{min} = 60\%,$$

Якщо величина $\text{Е}_{\text{в}} > \tau_{min}$, то потенційний інвестор може бути зацікавлений у фінансуванні впровадження науково-технічної розробки та виведенні її на ринок, тобто в її комерціалізації [41].

Далі розраховуємо період окупності інвестицій $\text{Т}_{\text{ок}}$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_в}\,, \tag{4.23}$$

де $\text{Е}_{\text{в}}$ – внутрішня економічна дохідність вкладених інвестицій.

$$\text{Т}_{\text{ок}} = \frac{1}{1{,}6} = 0{,}38 \text{ року}.$$

Якщо $\text{Т}_{\text{ок}}$ < 3-х років, то це свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження цієї розробки та виведення її на ринок.

### 4.5 Висновки до розділу

У цьому розділі проведено оцінювання комерційного потенціалу розробки програмного продукту для захищеної системи оповіщення населення про ракетну небезпеку в автоматичному та "ручному" режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифрування даних. Технологічний аудит було здійснено за участю трьох незалежних експертів, які дійшли висновку, що рівень комерційного потенціалу розробки перевищує середні показники.

Під час оцінювання встановлено, що розробка є якісною та конкурентоспроможною. За результатами аналізу, рівень комерційного потенціалу становить 48,3, що відповідає категорії "високий". Розрахунки витрат на науково-дослідну, дослідно-конструкторську та конструкторсько-технологічну роботу показали, що загальні витрати на розробку складають 861 950,8 грн. Розрахована абсолютна ефективність вкладених інвестицій у сумі 125 848 367,1 грн свідчить про те, що інвестор може розраховувати на значний прибуток від комерціалізації розробленого програмного продукту.

Щорічна ефективність вкладених інвестицій у наукову розробку становить 260%, що перевищує мінімальну бар'єрну ставку дисконтування у 40%. Це вказує на значний інтерес інвесторів до фінансування даного проекту. Термін окупності вкладених інвестицій у реалізацію проекту становить 0,38 року, що також підтверджує доцільність фінансування цієї розробки.

Отже, аналізуючи отримані економічні показники, можна зробити висновок, що запропонована розробка програмного продукту має високий комерційний потенціал і, відповідно, є обґрунтованою для подальшого впровадження.

# ВИСНОВКИ

У цій роботі було проведено дослідження та розроблено захищену систему оповіщення населення про ракетну небезпеку, що працює в автоматичному та «ручному» режимах, з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних.

У процесі дослідження було встановлено, що захищені системи оповіщення є критично важливим елементом інфраструктури у сучасному світі, особливо в умовах зростаючої загрози ракетних ударів. Використання сучасних технологій, таких як односторонній зв'язок, резервні канали обміну та шифрування даних, дозволяє забезпечити високий рівень надійності, конфіденційності та швидкості оповіщення населення про надзвичайні ситуації.

Процес розробки системи включав в себе аналіз теоретичних основ захисту інформації, проектування резервних каналів обміну, розробку схеми та алгоритму роботи пристрою оповіщення, програмну реалізацію серверної та апаратної частин, а також економічне обґрунтування розробки.

Отримані результати демонструють, що розроблена система є ефективним засобом забезпечення безпеки населення в умовах ракетної небезпеки. Вона може бути успішно впроваджена як на рівні державних структур, так і у приватних компаніях та організаціях для забезпечення захисту життя та майна громадян у випадку виникнення надзвичайних ситуацій.

Загальні висновки свідчать про важливість подальшого дослідження та розвитку захищених систем оповіщення, оскільки це дозволить забезпечити ще більшу безпеку населення в умовах постійно зростаючих загроз.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. AVSystem - Shaping the world of connected devices. AVSystem – Shaping The World of Connected Devices. URL: https://www.avsystem.com/coiote-iot-device-management-platform/iot-device-management/.

2. BasuMallick C. What Is IoT Device Management? Definition, Key Features, and Software - Spiceworks. Spiceworks Inc. URL: https://www.spiceworks.com/tech/iot/articles/what-is-iot-device-management/.

3. Gillis A. S. What is IoT (Internet of Things) and How Does it Work? | Definition from TechTarget. IoT Agenda. URL: https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT.

4. IoT Device Management_IoT_IoTDM-HUAWEI CLOUD. Huawei Cloud. URL: https://www.huaweicloud.com/intl/en-us/product/iot.html.

5. Revolutionary IoT Device Management | Shaping the future IoT. Novatec Consulting GmbH. URL: https://www.novatec-gmbh.de/en/insights/blog/iot-device-management/.

6. thingsboard. ⊳ IoT Device Management Platform. ThingsBoard. URL: https://thingsboard.io/device-management/.

7. What is IoT Device Management? | Definition & Benefits | Software AG. Software AG. URL: https://www.softwareag.com/en_corporate/resources/iot/article/iot-device-management.html.

8. What is IoT Device Management? | NinjaOne. NinjaOne. URL: https://www.ninjaone.com/blog/what-is-iot-device-management/.

9. What is IoT? - Internet of Things Explained - Amazon AWS. Amazon Web Services, Inc. URL: https://aws.amazon.com/ru/what-is/iot/#:~:text=The%20term%20IoT,%20or%20Internet,as%20between%20the%20devices%20themselves.

10. What is IoT? The internet of things explained. Network World. URL: https://www.networkworld.com/article/963923/what-is-iot-the-internet-of-things-explained.html.

11. What is the Internet of Things (IoT)? | IBM. IBM - United States. URL: https://www.ibm.com/topics/internet-of-things.

12. What is the Internet of Things (IoT)?. Oracle | Cloud Applications and Cloud Platform. URL: https://www.oracle.com/jo/internet-of-things/what-is-iot/.

13. What Is the Internet of Things (IoT)? With Examples. Coursera. URL: https://www.coursera.org/articles/internet-of-things.

14. What is the IoT? Everything you need to know about the Internet of Things right now. ZDNET. URL: https://www.zdnet.com/article/what-is-the-internet-of-things-everything-you-need-to-know-about-the-iot-right-now/.

15. Cloudflare Application Services | Security and Performance. Cloudflare. URL: https://www.cloudflare.com/application-services/?utm_source=google&utm_medium=cpc&utm_campaign=DG_EMEig7BqZ7VtZrN6wUzk1Nhp6nkaAgwEEALw_wcB&gclsrc=aw.ds).

16. Кібербезпека. URL: https://www.span.eu/ua/рішення-та-послуги/сервіси-з-безпеки/кібербезпека/?gad_source=1&gclid=Cj0KCQiA67CrBhC1ARIsACKAa8REXQS8JOs2ZaaOW6g5OyaeVWM8ksz3M6viwjX_pclBefnGsQgibfMaAgYHEALw_wcB.

17. Contributors to Wikimedia projects. Information security - Wikipedia. Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Information_security

18. Imperva. URL: https://www.imperva.com/learn/data-security/information-security-infosec/.

19. What is information security (infosec)?. Cisco. URL: https://www.cisco.com/c/en/us/products/security/what-is-information-security-infosec.html.

20. What is information security (infosec)? Goals, types and applications. Exabeam. URL: https://www.exabeam.com/explainers/information-security/information-security-goals-types-and-applications/.

21. Yasar K., Wright G., Teravainen T. What is information security (infosec)? – techtarget definition. Security. URL: https://www.techtarget.com/searchsecurity/definition/information-security-infosec.

22. What is information security? | IBM. IBM in Deutschland, Österreich und der Schweiz | IBM. URL: https://www.ibm.com/topics/information-security.

23. What is information security? - geeksforgeeks. GeeksforGeeks. URL: https://www.geeksforgeeks.org/what-is-information-security/.

24. INFOSEC - Glossary | CSRC. NIST Computer Security Resource Center | CSRC. URL: https://csrc.nist.gov/glossary/term/INFOSEC.

25. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. Вінниця : ВНТУ, 2021. 42 с.

26. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа. Вінниця : ВНТУ, 2016. 113 с.

27. What is authorization? - examples and definition - auth0. *Auth0*. URL: https://auth0.com/intro-to-iam/what-is-authorization .

28. Academy B. Seed Phrase | Binance Academy. Binance Academy. URL: https://academy.binance.com/en/glossary/seed-phrase.

29. What is authorization? - examples and definition - auth0. *Auth0*. URL: https://auth0.com/intro-to-iam/what-is-authorization .

30. What is javascript? A basic introduction to JS for beginners. Hostinger Tutorials. URL: https://www.hostinger.com/tutorials/what-is-javascript).

31. Visual studio code. Visual Studio Code. URL: https://code.visualstudio.com/).

32. Редактор коду visual studio code. Habr. URL: https://habr.com/ua/articles/490754

33. JavaScript. URL: https://developer.mozilla.org/en-US/docs/Web/JavaScript.

34. Electron. Electron. URL: https://www.electronjs.org/ .

35. Node.js. Node.js. URL: https://nodejs.org/uk .

36. The Modern JavaScript. URL: https://javascript.info/ .

37. Sufiyan T. What is node.js: a comprehensive guide. *Simplilearn.com*. URL: https://www.simplilearn.com/tutorials/nodejs-tutorial/what-is-nodejs ).

38. Megida D. What is javascript? A definition of the JS programming language. freeCodeCamp.org. URL: https://www.freecodecamp.org/news/what-is-javascript-definition-of-js/

39. What is javascript? A basic introduction to JS for beginners. Hostinger Tutorials. URL: https://www.hostinger.com/tutorials/what-is-javascript).

40. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. Вінниця : ВНТУ, 2021. 42 с.

41. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа. Вінниця : ВНТУ, 2016. 113 с

# ДОДАТКИ

**Додаток А. Технічне завдання**

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

**ЗАТВЕРДЖУЮ**
Голова секції "Управління інформаційною безпекою" кафедри МБІС
д.т.н., професор
_____ Юрій ЯРЕМЧУК
"____" _____ 2024 р.

# ТЕХНІЧНЕ ЗАВДАННЯ

до магістерської кваліфікаційної роботи на тему:
Захищена система оповіщення населення про ракетну небезпеку в автоматичному та "ручному" режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних
08-72.МКР.000.00.000.ТЗ

Керівник магістерської кваліфікаційної роботи
д.т.н., професор каф. МБІС
_____ Яремчук Ю.Є.

Вінниця – 2024 р.

**1. Найменування та область застосування**

Захищена система оповіщення населення про ракетну небезпеку в автоматичному та “ручному” режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних **2.Підстава для розробки**

Розробка виконується на основі наказу ректора ВНТУ № 81 від 11. 03. 2024 р.

**3. Мета та призначення розробки**

3.1 Мета розробки: реалізація захищеної системи оповіщення населення про ракетну небезпеку в автоматичному та “ручному” режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних

3.2 Призначення: реалізація захищеної системи оповіщення населення про ракетну небезпеку

**4. Джерела розробки**

4.1. Ахрамович В. М. Ідентифікація й аутентифікація, керування доступом // Сучасний захист інформації. – 2016. №4.– С. 47-51.

4.2. Бурячок В.Л. Політика інформаційної безпеки: підручник. / В.Л.Бурячок, Р.В.Грищук, В.О.Хорошко / За заг. ред. докт. техн. наук, проф. В.О. Хорошка. – К.: ПВП «Задруга», 2014. – 222 с.

4.3. Єсін В.І. Безпека інформаційних систем і технологій / В.І.Єсін, О.О. Кузнецов, Л.С. Сорока. – Харків: ХНУ імені В.Н. Каразіна, 2013. – 632 с.

4.4. ZakariaOmar, ZangooeiToomaj, MohdAfiziMohdShukran. Enhancing Mixing Recognition-Based and Recall-Based Approach in Graphical Password Scheme. IJACT, Vol. 4, No. 15, pp. 189-197, 2012.

**5. Вимоги до програми**

5.1 Вимоги до функціональних характеристик:

5.1.1 Програмний засіб повинен мати зручний, легкий у використанні інтерфейс користувача;

5.1.2 Реалізація не повинна вимагати спеціальних ліцензійних програмних додатків;

5.1.3 Програмний засіб повинен виконувати процес автентифікації користувачів у системі.

5.2 Вимоги до надійності:

5.2.1 Програмний засіб повинен працювати без помилок, у випадку виникнення критичних ситуацій необхідно передбачити виведення відповідних повідомлень;

5.2.2 Бази даних повинні бути налаштовані на автоматичне створення резервних копій;

5.2.3 Програмний засіб повинен виконувати свої функції.

5.3 Вимоги до складу і параметрів технічних засобів:

– процесор – Pentium 1500 МГц і подібні до них;

– оперативна пам’ять – не менше 512 Mb;

– середовище функціонування – операційна система сімейство Windows;

– вимоги до техніки безпеки при роботі з програмою повинні відповідати існуючим вимогам та стандартам з техніки безпеки при користуванні комп’ютерною технікою.

**6. Вимоги до програмної документації**

6.1 Обов'язкова поетапна інструкція для майбутніх користувачів, наведена у пункті 3.4

**7. Вимоги до технічного захисту інформації**

7.1 Необхідно забезпечити захист розроблюваного програмного засобу від несанкціонованого використання.

7.2 Неможливість отримання доступу незареєстрованих користувачів до інформаційних ресурсів.

**8. Техніко-економічні показники**

8.1 Цінність результатів використання даного проекту повинна перевищувати витрати на його реалізацію.

8.2 Має бути реалізований таким чином, щоб підходити для використання широкого загалу.

**9. Стадії та етапи розробки**

| № | Назва етапів магістерської кваліфікаційної роботи | Строк виконання етапів роботи | | Примітка |
|---|---|---|---|---|
| 1. | Визначення напрямку магістерської роботи, формулювання теми | 1.03.2024 | 15.03.2024 | |
| 2. | Аналіз предметної області обраної теми | 15.03.2024 | 1.04.2024 | |
| 3. | Розробка роботи | 1.04.2024 | 10.04.2024 | |
| 4. | Написання магістерської роботи на основі розробленої теми | 10.04.2024 | 20.04.2024 | |
| 5. | Передзахист магістерської кваліфікаційної роботи | 20.04.2024 | 10.05.2024 | |
| 6. | Виправлення, уточнення, корегування магістерської кваліфікаційної роботи | 10.05.2024 | 4.06.2024 | |
| 7. | Захист магістерської кваліфікаційної роботи | 10.06.2024 | 10.06.2024 | |

**10. Порядок контролю та прийому**

10.1 До приймання магістерської кваліфікаційної роботи надається:

– ПЗ до магістерської кваліфікаційної роботи;

– програмний додаток;

– презентація;

– відзив керівника роботи;

– відзив опонента

Технічне завдання до виконання прийняв _____ Пономарьов В.О.

# Додаток Б. Лістинг програми

```
#include <boost/asio.hpp>
#include <iostream>
#include <thread>
using boost::asio::ip::tcp;
void session(tcp::socket socket) {
   try {
      while (true) {
         char data[1024];
         boost::system::error_code error;
         size_t length = socket.read_some(boost::asio::buffer(data), error);
         if (error == boost::asio::error::eof) {
            break; // Connection closed cleanly by peer.
         } else if (error) {
            throw boost::system::system_error(error); // Some other error.
         std::cout << "Received: " << std::string(data, length) << std::endl;
   } catch (std::exception& e) {
      std::cerr << "Exception in thread: " << e.what() << std::endl;
int main() {
   try {
      boost::asio::io_context io_context;
class DataProcessor {

public:

   std::string process_data(const std::string& data); // Головний метод обробки даних

private:

   std::string analyze_data(const std::string& data); // Метод аналізу даних

   std::string create_alert_message(const std::string& alert_level); // Метод генерації
повідомлень

   int calculate_threat_level(const std::string& data); // Метод розрахунку рівня загрози

   bool validate_data(const std::string& data); // Метод валідації даних

};

std::string DataProcessor::analyze_data(const std::string& data) {
   int threat_level = calculate_threat_level(data);
   if (threat_level >= 80) {
      return "high";
   } else if (threat_level >= 50) {
      return "medium";
   } else {
      return "low";
   }
}
int DataProcessor::calculate_threat_level(const std::string& data) {

  int threat_level = 0;

  if (data.find("missile") != std::string::npos) {

    threat_level += 50; }

  if (data.find("alert") != std::string::npos) {

    threat_level += 30;}

  return threat_level;}

std::string DataProcessor::create_alert_message(const std::string& alert_level) {
```

```
    if (alert_level == "high") {
        return "Missile threat detected! Take immediate action!";
    } else if (alert_level == "medium") {
        return "Potential threat detected. Stay alert.";
    } else {
        return "No immediate threat detected.";
    }
}
#include <openssl/evp.h>
#include <openssl/aes.h>
#include <openssl/rand.h>
#include <openssl/hmac.h>
#include <iostream>
#include <string>
#include <vector>
class CryptoManager {
public:
    CryptoManager();
    ~CryptoManager();
    std::vector<unsigned char> encrypt(const std::string& plaintext);
    std::string decrypt(const std::vector<unsigned char>& ciphertext);
    std::vector<unsigned char> generate_hmac(const std::string& data);
    bool verify_hmac(const std::string& data, const std::vector<unsigned char>& hmac);
private:
    EVP_CIPHER_CTX* encrypt_ctx;
    EVP_CIPHER_CTX* decrypt_ctx;
    unsigned char key[32]; // 256-bit key
    unsigned char iv[16];  // 128-bit IV
    unsigned char hmac_key[32]; // 256-bit HMAC key
    void initialize
CryptoManager::CryptoManager() {
    encrypt_ctx = EVP_CIPHER_CTX_new();
    decrypt_ctx = EVP_CIPHER_CTX_new();
    initialize_keys();
}
CryptoManager::~CryptoManager() {
    EVP_CIPHER_CTX_free(encrypt_ctx);
    EVP_CIPHER_CTX_free(decrypt_ctx);
}
void CryptoManager::initialize_keys() {
    // Генерація ключа та IV для AES
    RAND_bytes(key, sizeof(key));
```

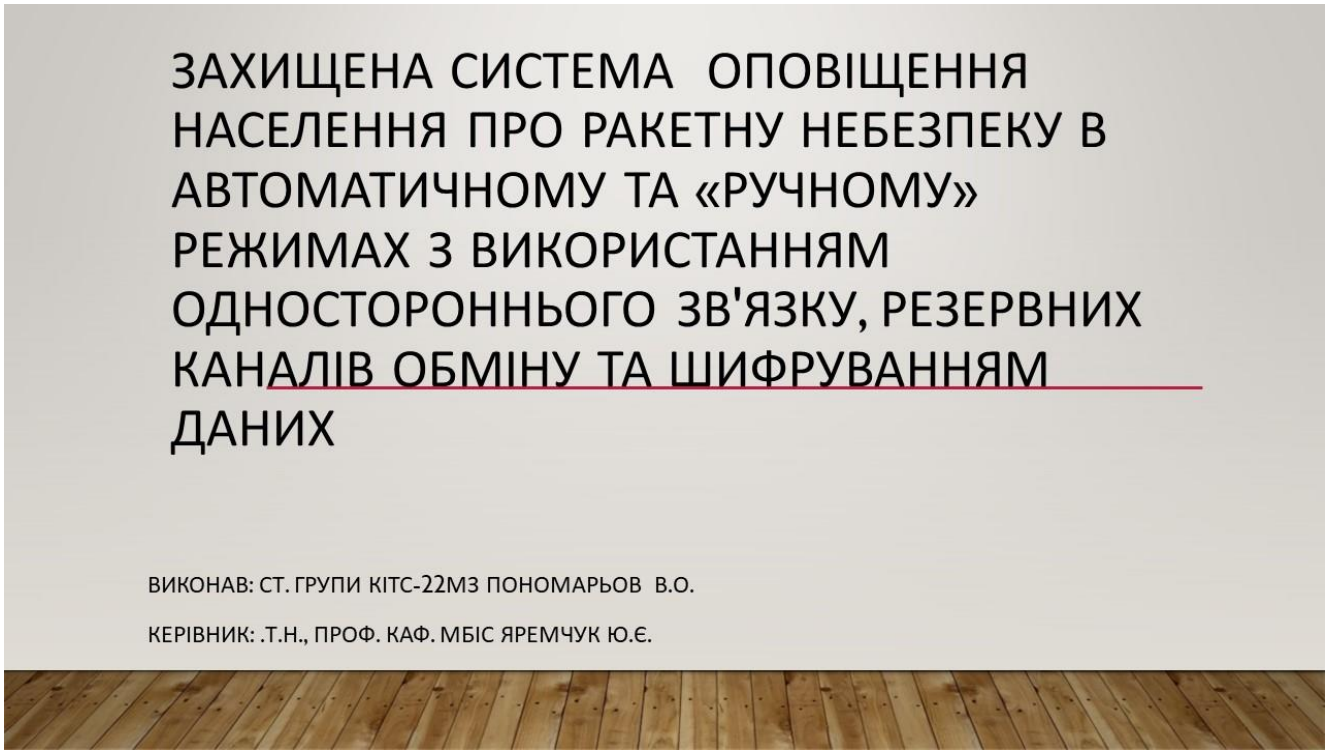
```
        RAND_bytes(iv, sizeof(iv));
        // Генерація ключа для HMAC
        RAND_bytes(hmac_key, sizeof(hmac_key));
    }
    std::vector<unsigned char> CryptoManager::encrypt(const std::string& plaintext) {
        std::vector<unsigned char> ciphertext(plaintext.size() + AES_BLOCK_SIZE);
        int len;
        int ciphertext_len;
        EVP_EncryptInit_ex(encrypt_ctx, EVP_aes_256_cbc(), NULL, key, iv);
        EVP_EncryptUpdate(encrypt_ctx, ciphertext.data(), &len, reinterpret_cast<const unsigned
char*>(plaintext.c_str()), plaintext.length());
        ciphertext_len = len;
        EVP_EncryptFinal_ex(encrypt_ctx, ciphertext.data() + len, &len);
        ciphertext_len += len;

        ciphertext.resize(ciphertext_len);
        return ciphertext;
    }
    #include <mysql_driver.h>
    #include <mysql_connection.h>
    #include <cppconn/driver.h>
    #include <cppconn/exception.h>
    #include <cppconn/resultset.h>
    #include <cppconn/statement.h>
    #include <cppconn/prepared_statement.h>
    #include <string>
    #include <vector>
    class DatabaseManager {
    public:
        DatabaseManager(const std::string& db_host, const std::string& db_user, const std::string& db_password,
const std::string& db_name);
        ~DatabaseManager();
        void connect();
        void disconnect();
        void insert_event(const std::string& event_type, const std::string& event_description);
        std::vector<std::pair<std::string, std::string>> get_events();
    private:
        sql::Driver* driver;
        sql::Connection* connection;
        std::string db_host;
        std::string db_user;
        std::string db_password;
        std::string db_name;
    };
```

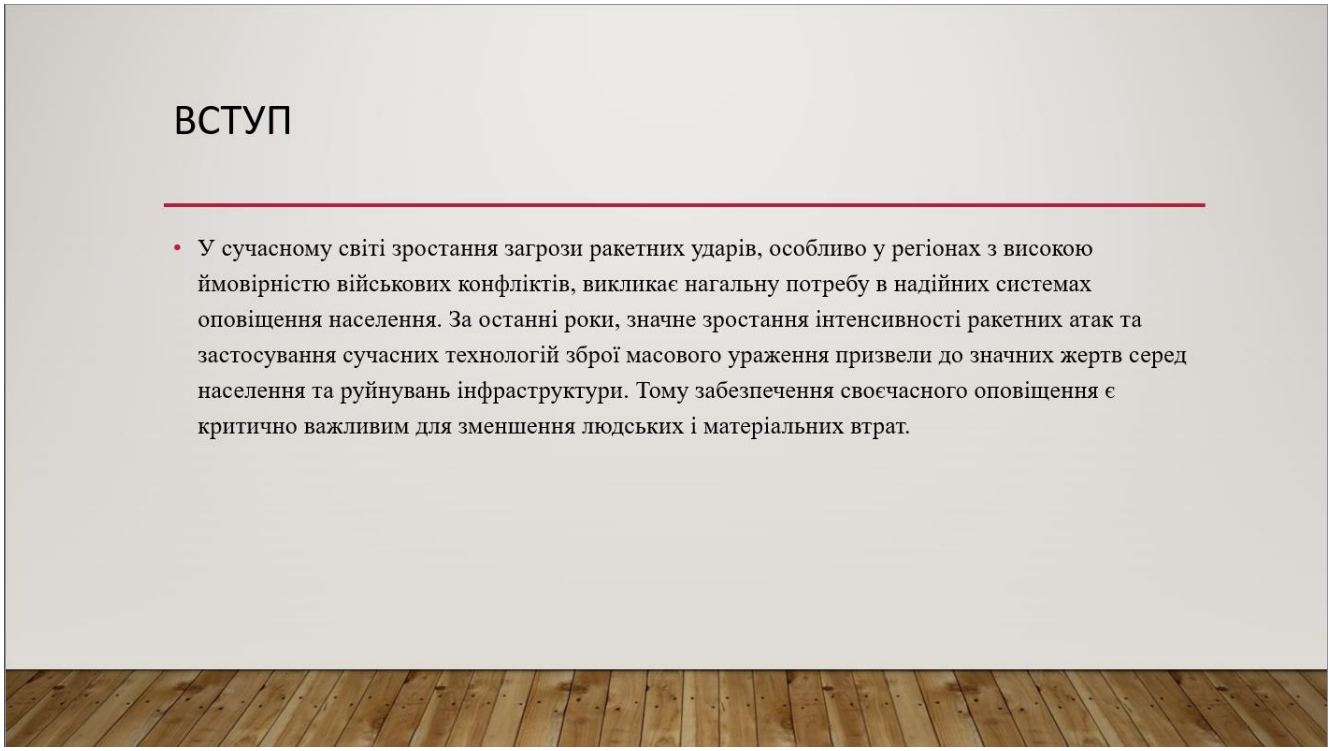
```
#include <SPI.h>
#include <Ethernet.h>
// Встановлення апаратних пінів для сирени та світлового сигналу
const int sirenPin = 8;
const int lightPin = 9;
// MAC-адреса та IP-адреса Ethernet щита
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192, 168, 1, 177);
IPAddress server(192, 168, 1, 2); // IP-адреса сервера
EthernetClient client;
void setup() {
  // Налаштування пінів
  pinMode(sirenPin, OUTPUT);
  pinMode(lightPin, OUTPUT);
  // Ініціалізація Ethernet з'єднання
  Ethernet.begin(mac, ip);
  Serial.begin(9600);
  // Затримка для стабілізації з'єднання
  delay(1000);
}
void loop() {
  // Підключення до сервера
  if (client.connect(server, 80)) {
    // Запит до сервера для отримання команди
    client.println("GET /get_command HTTP/1.1");
    client.println("Host: 192.168.1.2");
    client.println("Connection: close");
    client.println();
    // Очікування відповіді від сервера
    while (client.connected()) {
      if (client.available()) {
        String line = client.readStringUntil('\n');
        if (line == "\r") {
          break;
        }
      }
    }
    // Отримання команди від сервера
    String command = client.readStringUntil('\n');
    Serial.println("Received command: " + command);
    // Виконання команди в залежності від режиму
    if (command == "ACTIVATE_SIREN") {
      digitalWrite(sirenPin, HIGH);  // Активувати сирену
    } else if (command == "DEACTIVATE_SIREN") {
      digitalWrite(sirenPin, LOW);   // Деактивувати сирену
    } else if (command == "ACTIVATE_LIGHT") {
      digitalWrite(lightPin, HIGH);  // Активувати світловий сигнал
    } else if (command == "DEACTIVATE_LIGHT") {
      digitalWrite(lightPin, LOW);   // Деактивувати світловий сигнал
    }
    // Закриття з'єднання з сервером
    client.stop();
  }
#include <SPI.h>

#include <Ethernet.h>

#include <SoftwareSerial.h>

const int sirenPin = 8;
```

```
const int lightPin = 9;
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192, 168, 1, 177);
IPAddress server(192, 168, 1, 2); // IP-адреса сервера
SoftwareSerial gsmSerial(10, 11); // RX, TX піні для GSM модуля
const char* apn = "your_apn"; // APN для підключення до мобільного інтернету
const char* gprsUser = "user"; // ім'я користувача
const char* gprsPass = "password"; // пароль
EthernetClient ethernetClient;
bool isMainChannelActive = true;

void setup() {
  pinMode(sirenPin, OUTPUT);
  pinMode(lightPin, OUTPUT);
  // Ініціалізація Ethernet з'єднання
  Ethernet.begin(mac, ip);
  Serial.begin(9600);
  // Ініціалізація GSM модуля
  gsmSerial.begin(9600);
  initializeGSM();
  // Затримка для стабілізації з'єднання
  delay(1000);
}
void loop() {
  if (isMainChannelActive) {
    if (!connectToServer(ethernetClient)) {
      Serial.println("Main channel failed, switching to backup.");
      isMainChannelActive = false;
  // Затримка перед наступною ітерацією
  delay(1000);
}
bool connectToServer(Client &client) {
  if (client.connect(server, 80)) {
    // Запит до сервера для отримання команди
    client.println("GET /get_command HTTP/1.1");
    client.println("Host: 192.168.1.2");
    client.println("Connection: close");
    client.println();
    // Очікування відповіді від сервера
```

```
    while (client.connected()) {
      if (client.available()) {
        String line = client.readStringUntil('\n');
        if (line == "\r") {
          break;
    // Отримання команди від сервера
    String command = client.readStringUntil('\n');
    Serial.println("Received command: " + command);
    if (command == "ACTIVATE_SIREN") {
      digitalWrite(sirenPin, HIGH);  // Активувати сирену
    } else if (command == "DEACTIVATE_SIREN") {
      digitalWrite(sirenPin, LOW);
    } else if (command == "ACTIVATE_LIGHT") {
      digitalWrite(lightPin, HIGH);  // Активувати світловий сигнал
    } else if (command == "DEACTIVATE_LIGHT") {
      digitalWrite(lightPin, LOW);   // Деактивувати світловий сигнал
    }
    client.stop();
    return true;
  } else {
    return false;
void initializeGSM() {
  gsmSerial.println("AT");
  delay(1000);
  gsmSerial.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"");
  delay(1000);
  gsmSerial.println("AT+SAPBR=3,1,\"APN\",\"" + String(apn) + "\"");
  delay(1000);
  gsmSerial.println("AT+SAPBR=1,1");
  delay(1000);
  gsmSerial.println("AT+SAPBR=2,1");
  delay(1000);
}
```

**Додаток В. Ілюстративний матеріал**



## АКТУАЛЬНІСТЬ ТА МЕТА

- Актуальність. Захищені системи оповіщення, які здатні працювати в автоматичному та «ручному» режимах, є важливими для забезпечення безпеки як в мирний час, так і в умовах війни. Використання одностороннього зв'язку унеможливлює перехоплення і модифікацію повідомлень з боку противника, що є особливо важливим у військових умовах
- Метою даної роботи є розробка захищеної системи оповіщення населення про ракетну небезпеку, що функціонує в автоматичному та «ручному» режимах, з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних.

## ВАЖЛИВІСТЬ ВИКОРИСТАННЯ ТА ВИДИ СИСТЕМ ОПОВІЩЕННЯ НАСЕЛЕННЯ

- Системи оповіщення населення мають вирішальне значення для забезпечення безпеки та захисту громадян у випадку надзвичайних ситуацій, таких як ракетна небезпека. Ці системи надають можливість швидкого та ефективного сповіщення громадян про потенційні загрози та надають необхідні інструкції для забезпечення їх власної безпеки.
- Існують різноманітні види систем оповіщення населення, включаючи:
  - системи автоматизованого масового повідомлення (САМП);
  - системи електронних табло та попередження на дорогах;
  - системи публічного оповіщення;
  - системи радіо- та телевізійного оповіщення;

# РЕЗЕРВНІ КАНАЛИ ЗВ'ЯЗКУ

- Резервні канали зв'язку - це альтернативні способи комунікації, які використовуються, коли основний канал зв'язку недоступний.
- Існує багато видів резервних каналів зв'язку, таких як:

– мобільний зв'язок: Мобільні телефони можуть використовуватися для голосового зв'язку, текстових повідомлень та доступу до Інтернету;

– супутниковий зв'язок: Супутниковий зв'язок може використовуватися для голосового зв'язку, текстових повідомлень та доступу до Інтернету в районах, де недоступний мобільний зв'язок;

– радіо: Радіозв'язок може використовуватися для голосового зв'язку в районах, де недоступний мобільний зв'язок або супутниковий зв'язок;

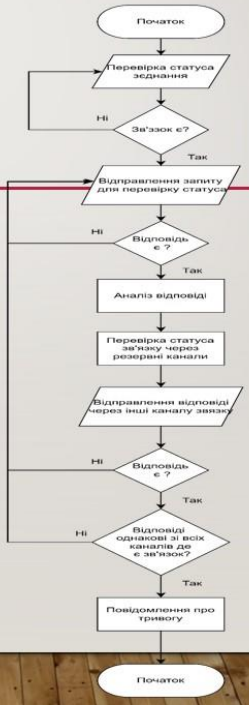
– сигнали бедствия: Сигнали бедствия, такі як димові шашки або сигнальні пістолети, можуть використовуватися для подачі сигналу про допомогу в надзвичайних ситуаціях;

– мережі радіочастотного зв'язку: Мережі радіочастотного зв'язку, такі як DMR, можуть використовуватися для голосового зв'язку та передачі даних;

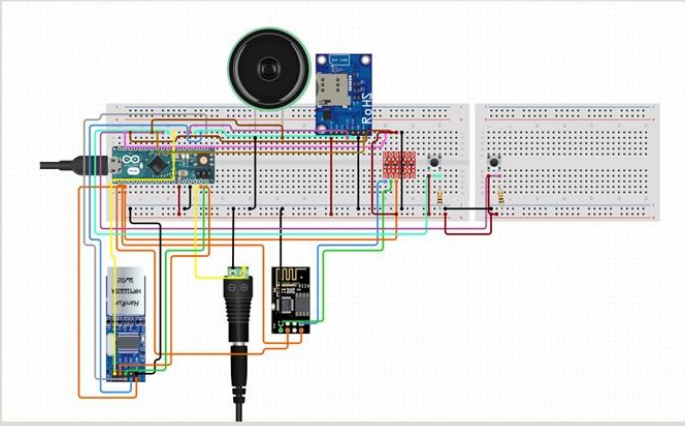
# АНАЛІЗ ВИДІВ ЗВ'ЯЗКІВ ДЛЯ ОБМІНУ ІНФОРМАЦІЄЮ

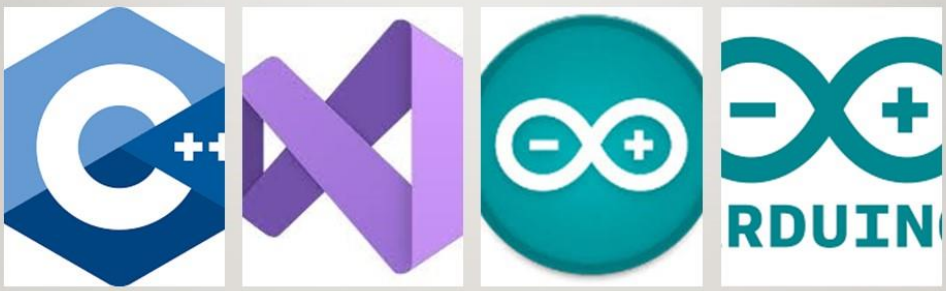
| Вид зв'язку | Швидкість і доставка | Надійність | Покриття і доступність | Вартість | Зручність та доступність для користувачів | Безпека | Можливості персоналізації та спрямованості |
|---|---|---|---|---|---|---|---|
| Мобільні додатки та SMS | Висока | Висока | Велика | Низька | Висока | Висока | Середня |
| Електронна пошта | Помірна | Висока | Глобальна | Низька | Висока | Середня | Низька |
| Соціальні мережі | Висока | Помірна | Глобальна | Низька | Висока | Середня | Висока |
| Автоматизовані голосові дзвінки | Висока | Висока | Локальна | Середня | Висока | Висока | Низька |
| Публічні табло та гучномовці | Низька | Висока | Локальна | Середня | Висока | Висока | Середня |
| Радіо та телевізійне мовлення | Висока | Висока | Глобальна | Висока | Висока | Середня | Середня |
| Мережа Wi-Fi | Висока | Помірна | Обмежена | Висока | Висока | Середня | Висока |

## АЛГОРИТМ РОБОТИ ПРИСТРОЮ ОПОВІЩЕННЯ НАСЕЛЕННЯ ПРО РАКЕТНУ НЕБЕЗПЕКУ



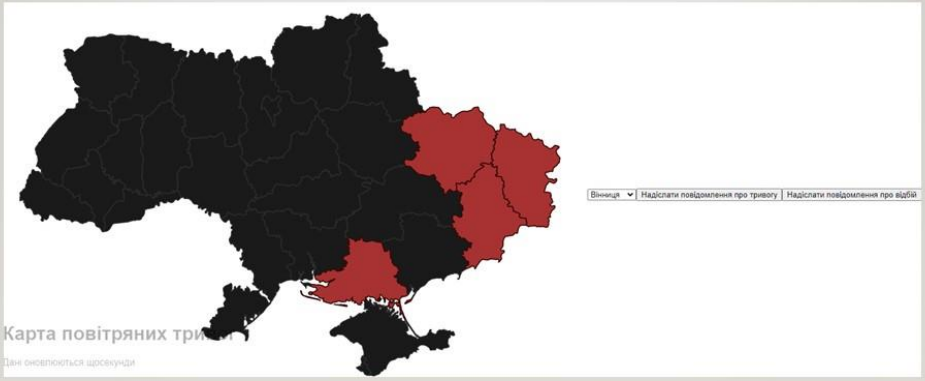

## РОЗРОБЛЕНА СХЕМА ПРИСТРОЮ ОПОВІЩЕННЯ НАСЕЛЕННЯ З РЕЗЕРВНИМИ КАНАЛАМИ ЗВ'ЯЗКУ



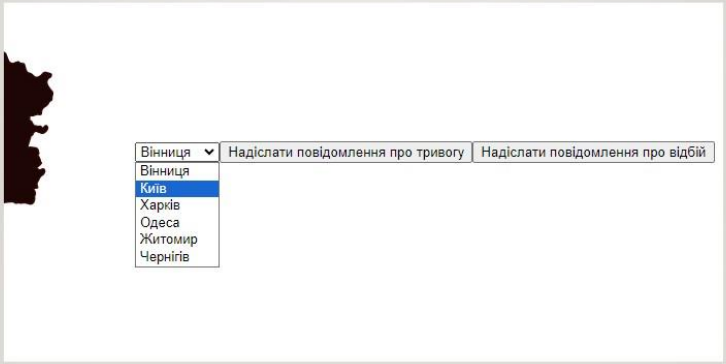


## ВИКОРИСТАНІ ТЕХНОЛОГІЇ

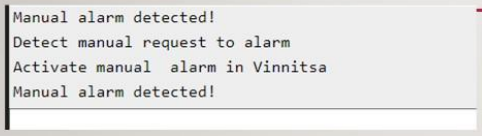
## КАРТА ВІДОБРАЖЕННЯ ТРИВОГ

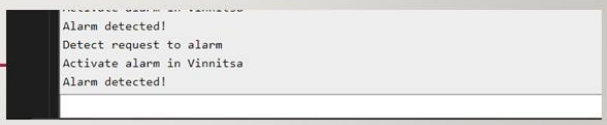


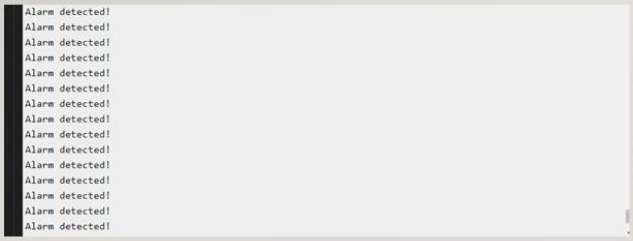
# НАДСИЛАННЯ ПОВІДОМЛЕННЯ ПРО ТРИВОГУ В РУЧНОМУ РЕЖИМІ

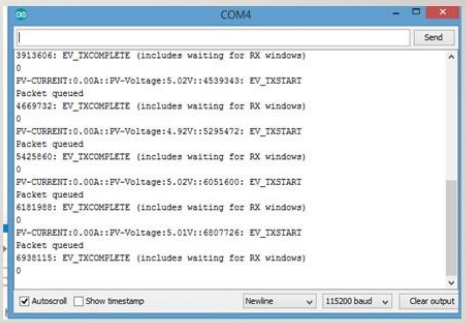


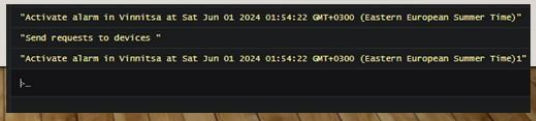

# ТЕСТУВАННЯ СИСТЕМИ



## ВИСНОВКИ

- Отже, процес розробки системи включав в себе аналіз теоретичних основ захисту інформації, проектування резервних каналів обміну, розробку схеми та алгоритму роботи пристрою оповіщення, програмну реалізацію серверної та апаратної частин, а також економічне обґрунтування розробки.
- Отримані результати демонструють, що розроблена система є ефективним засобом забезпечення безпеки населення в умовах ракетної небезпеки. Вона може бути успішно впроваджена як на рівні державних структур, так і у приватних компаніях та організаціях для забезпечення захисту життя та майна громадян у випадку виникнення надзвичайних ситуацій.

## ДЯКУЮ ЗА УВАГУ!

ПРОТОКОЛ
ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ
ЗАПОЗИЧЕНЬ

Назва роботи: Захищена система оповіщення населення про ракетну небезпеку в автоматичному та "ручному" режимах з використанням одностороннього зв'язку, резервних каналів обміну та шифруванням даних

Тип роботи: магістерська кваліфікаційна робота
(БДР, МКР)

Підрозділ: Кафедра менеджменту та безпеки інформаційних систем
Факультет менеджменту та інформаційної безпеки
(кафедра, факультет)

**Показники звіту подібності Unicheck**

Оригінальність 96 % Схожість 4 %

Аналіз звіту подібності (відмітити потрібне):
**1. Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.**
2. Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
3. Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Коваль Н.П.
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи _____ Пономарьов В.О.
(підпис) (прізвище, ініціали)

Керівник роботи _____ Яремчук Ю.Є.
(підпис) (прізвище, ініціали)