Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:

«Захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону»

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

Скомаровський В.В.
(прізвище та ініціали)

Керівник: к.т.н., проф., проф. каф. МБІС
Азарова А.О.
(прізвище та ініціали)
«04» грудня 2023 р.

Опонент: к.т.н., ст. викл. каф. ОТ
Обертюх М.Р.
(прізвище та ініціали)
«04» грудня 2023 р.

Допущено до захисту
Голова секції УБ кафедри МБІС

Юрій ЯРЕМЧУК
“04” грудня 2023 р.

Вінниця ВНТУ – 2023 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**
“ 20 ” вересня 2023 р.

З А В Д А Н Н Я
на магістерську кваліфікаційну роботу студенту
Скомаровському Владиславу Володимировичу
(прізвище, ім`я, по-батькові)

1. Тема роботи:

«Захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону»
Керівник роботи: к т.н., проф. каф. МБІС Азарова Анжеліка Олексіївна
(прізвище, ім`я, по-батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від “18” вересня 2023 року № 247
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:
Для досягнення мети роботи було поставлено наступні задачі: дослідити потреби організацій та звичайних людей у засобах захисту інформації; проаналізувати поширені способи захисту інформації та визначити їх недоліки; У першому розділі було проведено аналіз критичних інфраструктур та методів аналізу безпеки; У другому розділі було здійснено розробку бази даних та її нормалізаці, також було здійснено розробку методу захисту інформаційного ресурсу; В третьому розділі було здійснено реалізацію бази даних, модулю захисту та системи аналізу безпеки інфраструктури

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

У дугому розділі магістерської кваліфікаційної роботи наведено 6 рисунків, у третьому розділі – 8 рисунків

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Основна частина			
I	Азарова А.О., к т.н., проф. каф. МБІС		
II	Азарова А.О., к т.н., проф. каф. МБІС		
III	Азарова А.О., к т.н., проф. каф. МБІС		
Економічна частина			
IV	Причепа І.В., к.е.н., доц. каф. ЕПВМ		

7. Дата видачі завдання 20 вересня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи		Примітка
1.	Визначення напрямку магістерської роботи, формулювання теми	20.09.2023	31.09.2023	
2.	Аналіз предметної області обраної теми	01.10.2023	15.10.2023	
3.	Розробка роботи	16.10.2023	26.10.2023	
4.	Написання магістерської роботи на основі розробленої теми	27.10.2023	15.11.2023	
5.	Передзахист магістерської кваліфікаційної роботи	16.11.2023	24.11.2023	
6.	Виправлення, уточнення, корегування магістерської кваліфікаційної роботи	27.11.2023	04.12.2023	
7.	Захист магістерської кваліфікаційної роботи	11.12.2023	17.12.2023	

Студент _____ Скомаровський В.В.

Керівник роботи _____ Азарова А.О.

АНОТАЦІЯ

УДК 004.56.5(043.2)

Скомаровський В. В. Розробка захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону. Магістерська кваліфікаційна робота зі спеціальності 125 – «Кібербезпека», освітня програма «Кібербезпека інформаційних технологій та систем». Вінниця: ВНТУ, 2023. 124 с.

На укр.мові. Бібліогр.: 43 назв; рис.: 22; табл. 11.

У магістерській кваліфікаційній роботі здійснено розробку захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону.

В першому розділі роботи проведено аналіз теоретичного матеріалу у вибраній галузі: досліджено актуальність, особливості та забезпечення безпеки об'єктів критичної інфраструктури в галузі хімічної інфраструктури, а також здійснено огляд методів системного аналізу безпеки об'єктів та аналіз сучасних засобів автентифікації користувачів.

У другому розділі дослідження розглянуті особливості розробки інформаційного ресурсу для аналізу безпеки хімічної інфраструктури та заходи забезпечення захисту розробленого ресурсу від несанкціонованого доступу. Проведена розробка бази даних консолідованого інформаційного ресурсу та проведена нормалізація відношень бази даних.

У третьому розділі дослідження виконано практичну реалізацію бази даних консолідованого інформаційного ресурсу для аналізу безпеки хімічної інфраструктури. Також реалізовано систему аналізу безпеки хімічної інфраструктури та розроблено програмний модуль для забезпечення її захисту.

У четвертому розділі роботи проведено аналіз економічної доцільності розробки, підкреслено високий комерційний потенціал та можливості подальшого впровадження розробленого інформаційного ресурсу.

Ключові слова: екологічна інфраструктура, критична інфраструктура, системний аналіз, консолідація, безпека, інформаційний ресурс.

ABSTRACT

Skomarovskyy V. V. Development of a secure consolidated information resource for the systemic analysis of the chemical infrastructure security of the region. Master's thesis in the specialty 125 - "Cybersecurity," educational program "Cybersecurity of Information Technologies and Systems." Vinnytsia: VNTU, 2023. 111 p.

In Ukrainian language. Bibliographer: 43 titles; figures: 22; tables: 11.

The master's qualification work involves the development of a secure consolidated information resource for the systemic analysis of the chemical infrastructure security of the region.

In the first chapter of the work, an analysis of theoretical material in the selected field is conducted: relevance, features, and security measures for critical infrastructure objects in the field of chemical infrastructure are explored, along with a review of methods for systemic analysis of object security and an analysis of modern user authentication tools.

The second chapter of the research describes the features of developing an information resource for the analysis of the security of chemical infrastructure and measures to ensure the protection of the developed resource from unauthorized access. The development of a database for a consolidated information resource and normalization of database relationships are also carried out.

In the third chapter of the research, the practical implementation of the consolidated information resource database is carried out, including the implementation of a security analysis system for the chemical infrastructure. A software module is developed to ensure its protection, and the results of the system's operation are investigated.

The fourth chapter of the work analyzes the economic feasibility of the development, emphasizing its high commercial potential and opportunities for further implementation of the developed information resource.

Keywords: environmental infrastructure, critical infrastructure, systemic analysis, consolidation, security, information resource.

ЗМІСТ

ВСТУП

Актуальність. Хімічна інфраструктура містить різноманітні об'єкти, від хімічних заводів і транспортування небезпечних речовин до зберігання та обробки хімічних відходів. Загрози та ризики для цих об'єктів можуть бути різноманітні, і важливо мати консолідований інформаційний ресурс, який охоплює всі аспекти безпеки.

Хімічна інфраструктура є основою промислового та економічного розвитку регіону. Вона забезпечує виробництво хімічних речовин, які використовуються в різних галузях промисловості, сільського господарства, будівництва та інших сферах.

Хімічна інфраструктура є вразливою до атак з різних причин. Зловмисники можуть атакувати хімічну інфраструктуру для отримання економічної вигоди, політичної мети або просто для завдавання шкоди.

таки на хімічну інфраструктуру можуть мати масштабні наслідки. Вони можуть призвести до забруднення навколишнього середовища, шкоди здоров'ю людей та навіть до людських жертв.

З кожним роком зростає кількість загроз для хімічної інфраструктури. Зловмисники використовують все більш досконалі методи атак, які можуть призвести до серйозних наслідків.

Наприклад, у 2022 році було повідомлено про серію атак на хімічну інфраструктуру України, в результаті яких були знищені або пошкоджені підприємства, які виробляють хімічні речовини. Ці атаки призвели до забруднення навколишнього середовища та шкоди здоров'ю людей.

Необхідність об'єднання зусиль для підвищення безпеки хімічної інфраструктури. Для підвищення безпеки хімічної інфраструктури необхідно об'єднати зусилля різних органів влади, підприємств та організацій.

Таким чином, розробка захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону є важливою задачею для підвищення безпеки хімічної інфраструктури регіону.

Мета і задачі дослідження. Мета даної роботи полягає у створенні та впровадженні захищеного консолідованого інформаційного ресурсу. Цей ресурс призначений для сприяння системному аналізу та ефективного забезпечення безпеки медичної інфраструктури в даному регіоні.

Задачами дослідження є:

– аналіз теоретичних аспектів безпеки хімічної інфраструктури, розгляд актуальності, особливостей та існуючих методів забезпечення безпеки хімічних об'єктів;

– створення бази даних та нормалізація відношень для забезпечення ефективного зберігання та обробки інформації;

– практична реалізація інформаційного ресурсу та системи аналізу безпеки, розробка програмного забезпечення для взаємодії з ресурсом та забезпечення аналізу стану безпеки об'єктів хімічної інфраструктури;

– тестування та аналіз результатів, перевірка ефективності розробленого інструментарію та аналіз його впливу на безпеку об'єктів;

– економічне обґрунтування впровадження розробленого ресурсу, визначення витрат та користі від використання інформаційного ресурсу для підвищення рівня безпеки.

Об’єкт дослідження – хімічна інфраструктура регіону.

Предмет дослідження – консолідований інформаційний ресурс для системного аналізу безпеки хімічної інфраструктури.

Новизна роботи: вперше розроблено захищений консолідований інформаційний ресурс аналізу безпеки хімічної інфраструктури регіону, що дозволяє забезпечити комплексний погляд на проблему та вдосконалити існуючі методи забезпечення безпеки.

Практична цінність: впровадження розробленого консолідованого інформаційного ресурсу для підвищення ефективності систем управління та моніторингу безпеки хімічних об'єктів регіону, що сприятиме загальному зростанню рівня безпеки інфраструктури.

1 ТЕОРЕТИЧНІ ЗАСАДИ СТВОРЕННЯ КОНСОЛІДОВАНОГО ІНФОРМАЦІЙНОГО РЕСУРСУ АНАЛІЗУ БЕЗПЕКИ ХІМІЧНОЇ ІНФРАСТРУКТУРИ

У першому розділі магістерської кваліфікаційної роботи розглядається аналіз забезпечення безпеки об'єктів критичної інфраструктури, а також аналіз хімічної інфраструктури, розглянутої як складову критичної інфраструктури, а також аналіз процесу консолідації інформації.

На основі цього аналізу сформовано висновки та здійснено постановку задач для подальших досліджень та розробки.

1.1 Актуальність, особливості та основні принципи забезпечення безпеки об'єктів критичної інфраструктури

Актуальність проблеми забезпечення безпеки об'єктів критичної інфраструктури в сучасному світі визначається рядом факторів, які обумовлюють необхідність ретельного вивчення та ефективного впровадження відповідних заходів. Спричиненням цього є не тільки зростання технологічного розвитку, але й поява нових викликів у сфері кібербезпеки, терористичних загроз, природних катастроф та інших небезпечних ситуацій.

Критична інфраструктура – це сукупність об'єктів, систем і мереж, які є життєво важливими для забезпечення життєдіяльності суспільства та держави. До неї відносяться об'єкти енергетичної, транспортної, інформаційної, хімічної, телекомунікаційної, комунальної та інших галузей.

Забезпечення безпеки об'єктів критичної інфраструктури є одним з найважливіших завдань національної безпеки. Порушення функціонування цих об'єктів може призвести до серйозних економічних, соціальних і політичних наслідків, включаючи:

- порушення життєзабезпечення населення;
- зупинку виробництва;
- розлад транспортної системи;

– порушення хімічної системи;

– збій у роботі інформаційних систем;

– дестабілізацію суспільно-політичної ситуації.

Актуальність забезпечення безпеки об'єктів критичної інфраструктури зростає в міру розвитку інформаційних технологій і зростання загроз кібербезпеці. Кібернетичні атаки можуть призвести до вимкнення енергопостачання, транспортних систем, фінансових систем, інформаційних систем тощо.

Актуальність забезпечення безпеки об'єктів критичної інфраструктури визначається наступними факторами:

– важливість об'єктів критичної інфраструктури для життєдіяльності суспільства та держави. Порушення функціонування цих об'єктів може призвести до серйозних наслідків, тому забезпечення їх безпеки є пріоритетним завданням;

Наприклад, порушення електропостачання може призвести до зупинки виробництва, транспортної системи, фінансових систем, інформаційних систем тощо. Порушення роботи транспортної системи може призвести до дефіциту товарів і послуг, порушення ділових відносин, зростання соціальної напруженості тощо. Порушення роботи хімічної системи може призвести до економічного колапсу. Порушення роботи інформаційних систем може призвести до витоку інформації, порушення координації дій органів державної влади, військових і правоохоронних органів тощо [1].

– комплексність об'єктів критичної інфраструктури. Ці об'єкти часто є складними інженерними системами, що містить різні компоненти, такі як обладнання, програмне забезпечення, персонал тощо. Це ускладнює забезпечення їх безпеки;

Наприклад, для забезпечення безпеки енергосистеми необхідно захистити від фізичних і кіберзагроз електростанції, підстанції, лінії електропередач, системи управління тощо. Для забезпечення безпеки транспортної системи необхідно захистити від фізичних і кіберзагроз залізничні станції, аеропорти,

морські порти, автомобільні дороги тощо. Для забезпечення безпеки хімічної системи необхідно захистити від фізичних і кіберзагроз банки, фінансові установи, системи електронних платежів тощо. Для забезпечення безпеки інформаційних систем необхідно захистити від фізичних і кіберзагроз сервери, мережі, програмне забезпечення тощо [2].

– широкий спектр загроз. Об'єкти критичної інфраструктури можуть бути атаковані з різних напрямків, включаючи фізичні, інформаційні, економічні та інші.

У світлі цих факторів забезпечення безпеки об'єктів критичної інфраструктури є одним з найважливіших завдань національної безпеки.

Ось деякі конкретні приклади актуальності забезпечення безпеки об'єктів критичної інфраструктури:

– у 2015 році в Україні сталася масштабна кібератака на енергетичні об'єкти. У результаті атаки було виведено з ладу частина енергосистеми країни, що призвело до відключення електропостачання в деяких регіонах;

– у 2017 році в США відбулася масштабна кібератака на сервери компанії Equifax. У результаті атаки було викрадено персональні дані понад 143 мільйонів американців;

– у 2022 році Росія здійснила повномасштабне вторгнення в Україну. У рамках вторгнення російські війська атакували низку об'єктів критичної інфраструктури України, включаючи об'єкти енергетичної, транспортної та інформаційної інфраструктури.

Ці приклади показують, що загрози безпеці об'єктів критичної інфраструктури є реальними та можуть призвести до серйозних наслідків. Тому забезпечення безпеки цих об'єктів є пріоритетним завданням для всіх країн світу [3].

Актуальність забезпечення безпеки об'єктів критичної інфраструктури є високою і зростає в міру розвитку інформаційних технологій і зростання загроз

кібербезпеці. Успішне вирішення цього завдання є запорукою національної безпеки України.

Важливість об'єктів критичної інфраструктури для життєдіяльності суспільства та держави можна розглядати з наступних аспектів:

– економічний аспект. Об'єкти критичної інфраструктури є основою для економічного розвитку країни. Вони забезпечують безперебійне функціонування промисловості, торгівлі, транспорту, фінансів та інших галузей економіки;

– соціальний аспект. Об'єкти критичної інфраструктури забезпечують життєзабезпечення населення. Вони забезпечують доступ до води, електроенергії, теплопостачання, зв'язку, транспортних послуг тощо;

– політичний аспект. Об'єкти критичної інфраструктури є важливим фактором національної безпеки. Вони забезпечують функціонування держави в умовах надзвичайних ситуацій та військового конфлікту.

Комплексність об'єктів критичної інфраструктури можна розглядати з наступних аспектів:

– технічний аспект. Об'єкти критичної інфраструктури часто являють собою складні інженерні системи, що містить різні компоненти, такі як обладнання, програмне забезпечення, персонал тощо.

– організаційний аспект. Об'єкти критичної інфраструктури часто є об'єктами комунальної власності або державної власності. Вони підпорядковуються різним органам влади та управління.

– правовий аспект. Об'єкти критичної інфраструктури підпадають під дію різних нормативно-правових актів, що регулюють їх діяльність.

Широкий спектр загроз, які можуть бути спрямовані на об'єкти критичної інфраструктури, можна розглядати з наступних аспектів:

– фізичні загрози містять такі дії, як тероризм, диверсія, крадіжка обладнання та матеріалів тощо;

– інформаційні загрози містять такі дії, як кібератаки, розголошення конфіденційної інформації тощо;

– економічні загрози містять такі дії, як шахрайство, економічна блокада тощо.

Особливості забезпечення безпеки об'єктів критичної інфраструктури обумовлюють необхідність комплексного і всебічного підходу до цього завдання. Забезпечення безпеки цих об'єктів є складним і багатогранним завданням, яке вимагає застосування сучасних технологій і методів [4].

Забезпечення безпеки об'єктів критичної інфраструктури є складним і багатогранним завданням, яке вимагає комплексного підходу та застосування сучасних технологій і методів.

Враховуючи актуальність та особливості забезпечення безпеки об'єктів критичної інфраструктури, можна виділити такі основні засади цього завдання:

– системність. Безпека об'єкта критичної інфраструктури повинна розглядатися як єдина система, що містить всі його компоненти;

– комплексність. Забезпечення безпеки об'єкта критичної інфраструктури має бути комплексним, тобто містити заходи фізичного, інформаційного, економічного та інших напрямків;

– проактивність. Заходи безпеки повинні спрямовуватися на запобігання можливим загрозам, а не лише на їх ліквідацію після їх реалізації;

– стабільність. Заходи безпеки повинні бути стабільними, тобто адаптованими до змін в об'єкті критичної інфраструктури та в середовищі його функціонування.

Системність забезпечення безпеки об'єкта критичної інфраструктури передбачає, що всі його компоненти, включаючи обладнання, програмне забезпечення, персонал тощо, повинні розглядатися як єдина система. Це означає, що заходи безпеки повинні бути розроблені та реалізовані таким чином, щоб забезпечити захист усіх компонентів системи.

Комплексність забезпечення безпеки об'єкта критичної інфраструктури передбачає, що заходи безпеки повинні містити заходи фізичного, інформаційного, економічного та інших напрямків. Це означає, що необхідно

розглядати всі можливі загрози та розробляти заходи безпеки, які будуть ефективними для їх протидії.

Проактивність забезпечення безпеки об'єкта критичної інфраструктури передбачає, що заходи безпеки повинні спрямовуватися на запобігання можливим загрозам, а не лише на їх ліквідацію після їх реалізації. Це означає, що необхідно проводити активні заходи щодо виявлення та усунення загроз, а не лише реагувати на їх реалізацію [5].

Стабільність забезпечення безпеки об'єкта критичної інфраструктури передбачає, що заходи безпеки повинні бути стабільними, тобто адаптованими до змін в об'єкті критичної інфраструктури та в середовищі його функціонування. Це означає, що заходи безпеки повинні регулярно переглядатися та оновлюватися відповідно до змін, що відбуваються.

Врахування актуальності та особливостей забезпечення безпеки об'єктів критичної інфраструктури призводить до необхідності застосування таких заходів, як:

– забезпечення фізичної безпеки об'єкта критичної інфраструктури, включаючи охорону, пропускний режим, систему відеоспостереження, систему контролю доступу тощо;

– забезпечення інформаційної безпеки об'єкта критичної інфраструктури, включаючи політику інформаційної безпеки, систему управління інформаційною безпекою, систему захисту від кібератак тощо;

– забезпечення економічної безпеки об'єкта критичної інфраструктури, включаючи систему управління ризиками, систему протидії економічним злочинам тощо.

Основні засади забезпечення безпеки об'єктів критичної інфраструктури є важливими для успішного вирішення цього завдання. Вони дозволяють забезпечити комплексний і всебічний захист об'єктів критичної інфраструктури від різних загроз, а також підвищити їх стійкість до атак.

1.2 Загальна характеристика сфери діяльності "хімічна інфраструктура", як частина критичної інфраструктури регіону

Хімічна інфраструктура, як частина критичної інфраструктури регіону, відіграє невід'ємну роль у забезпеченні життєво важливих функцій для суспільства та економіки. Ця галузь діяльності містить різноманітні об'єкти, такі як хімічні заводи, склади, транспортні системи, об'єкти зберігання та обробки хімічних речовин. Збалансоване та ефективне функціонування хімічної інфраструктури є критичним для забезпечення виробництва, енергетики, медицини, сільськогосподарського сектору та інших сфер економіки [7].

Однак, разом зі своєю важливістю, хімічна інфраструктура природно виставлена на різноманітні загрози, які можуть містити не тільки техногенні аварії, але й катастрофічні події, такі як викиди небезпечних речовин, терористичні атаки або кіберзагрози. Враховуючи ці аспекти, забезпечення безпеки хімічної інфраструктури стає питанням національної та міжнародної важливості.

Хімічна інфраструктура – це сукупність підприємств, організацій і установ, що здійснюють виробництво, переробку, транспортування, зберігання та використання хімічних речовин. Вона є важливою частиною критичної інфраструктури регіону, оскільки забезпечує виробництво продукції, необхідної для життєдіяльності населення та економіки [8].

Хімічна промисловість – це галузь промисловості, яка виробляє різноманітні хімічні речовини, продукти та матеріали. Вона є однією з найбільш важливих галузей промисловості, оскільки забезпечує виробництво продукції, що використовується в різних сферах життєдіяльності, включаючи:

- сільське господарство: добрива, пестициди, ветеринарні препарати;
- будівництво: цемент, будівельні матеріали, лакофарбові матеріали;
- машинобудування: пластмаси, каучуки, синтетичні волокна, фарби, лаки;
- енергетика: паливно-мастильні матеріали, кислоти, основи;

– медицина: лікарські препарати, вакцини, діагностичні препарати;

– побутова хімія: миючі засоби, засоби для чищення, косметичні засоби;

Хімічна промисловість є також однією з найбільш наукомістких галузей промисловості. Вона постійно розвивається і вдосконалюється, розробляючи нові технології виробництва хімічної продукції.

Хімічна промисловість поділяється на кілька основних секторів:

– основна хімія – виробництво основних органічних і неорганічних речовин, які використовуються як сировина для виробництва інших хімічних продуктів;

– напівфабрикати – виробництво хімічних продуктів, які використовуються для виробництва готової продукції;

– готова продукція – виробництво хімічних продуктів, які використовуються в різних сферах життєдіяльності.

Виробництво хімічної продукції може призводити до забруднення навколишнього середовища. До основних джерел забруднення хімічної промисловості відносяться:

– викиди в атмосферу, які містять шкідливі речовини, такі як оксиди азоту, оксиди сірки, діоксид вуглецю, діоксид сірки.

– відходи виробництва, які можуть містити токсичні речовини, такі як важкі метали, органічні сполуки.

– водокористування, яке може призводити до забруднення води.

Забезпечення екологічної безпеки хімічної промисловості є важливим завданням. Воно містить заходи щодо запобігання забрудненню навколишнього середовища, а також заходи щодо ліквідації наслідків такого забруднення [9].

Об'єкти хімічної інфраструктури, які входять до критичної інфраструктури, це ті, які забезпечують виробництво та розподіл хімічної продукції, що є життєво необхідною для функціонування економіки та суспільства. До таких об'єктів відносяться:

– виробництво промислового газу;

– виробництво добрив або азотистих сполук;
– виробництво пестицидів або інших агрохімічних продуктів;
– виробництво вибухових речовин;
– виробництво основних органічних хімічних речовин;
– виробництво основних неорганічних речовин.

Промисловий газ – це газ, який використовується для промислових цілей, таких як виробництво, опалення, кондиціювання повітря та інші. До промислових газів відносяться [10]:

– природний газ;
– вугільний газ;
– газоподібні продукти нафтопереробки;
– синтетичні гази.

Виробництво промислового газу є важливим сектором хімічної промисловості. Воно забезпечує енергією та сировиною для різних галузей промисловості.

Добрива – це речовини, які вносяться в ґрунт для підвищення його родючості. До добрив відносяться:

– азотні добрива;
– фосфорні добрива;
– калійні добрива;
– комплексні добрива.

Виробництво добрив є важливим сектором хімічної промисловості. Воно забезпечує сільськогосподарські підприємства необхідними речовинами для підвищення врожайності сільгоспкультур [11].

Пестициди – це речовини, які використовуються для захисту рослин від шкідників, хвороб та бур'янів. До пестицидів відносяться:

– інсектициди;
– фунгіциди;
– гербіциди;

– десиканти.

Виробництво пестицидів є важливим сектором хімічної промисловості. Воно забезпечує сільськогосподарські підприємства необхідними речовинами для захисту сільгоспкультур.

Вибухові речовини – це речовини, які при певних умовах (нагріванні, ударі, терті) зазнають швидкого самозаймання та вибухового перетворення. До вибухових речовин відносяться:

– вогнівки;
– піроксилін;
– тетрил;
– алюмінієвий пероксид.

Виробництво вибухових речовин є важливим сектором хімічної промисловості. Воно забезпечує військово-промисловий комплекс необхідними речовинами для виробництва зброї та боєприпасів.

Основні органічні хімічні речовини – це речовини, які використовуються як сировина для виробництва інших органічних речовин. До основних органічних хімічних речовин відносяться:

– метанол;
– етанол;
– ацетилен;
– аміак;
– хлор.

Виробництво основних органічних хімічних речовин є важливим сектором хімічної промисловості. Воно забезпечує інші галузі промисловості необхідною сировиною для виробництва різноманітних продуктів [12].

Основні неорганічні речовини – це речовини, які використовуються як сировина для виробництва інших неорганічних речовин. До основних неорганічних речовин відносяться:

– кислоти;

– основи;

– солі;

– карбіди;

– силікати.

Виробництво основних неорганічних речовин є важливим сектором хімічної промисловості. Воно забезпечує інші галузі промисловості необхідною сировиною для виробництва різноманітних продуктів.

Хімічна промисловість має ряд особливостей, які відрізняють її від інших галузей промисловості [13]:

– висока інтенсивність використання сировини та енергії. Хімічна промисловість є однією з найбільш ресурсомістких галузей промисловості;

– велика кількість небезпечних речовин. У виробництві хімічної продукції використовуються різні небезпечні речовини, такі як кислоти, основи, хімікати, вибухові речовини тощо;

– висока екологічна небезпека. Виробництво хімічної продукції може призводити до забруднення навколишнього середовища.

Забезпечення безпеки хімічної промисловості є важливим завданням. Воно містить заходи щодо запобігання аваріям, пов'язаним з використанням небезпечних речовин, а також заходи щодо ліквідації наслідків таких аварій.

1.3 Консолідація інформації для забезпечення безпеки хімічної інфраструктури

В сучасному світі, де хімічна інфраструктура виступає ключовим елементом економіки та забезпечує життєво важливі послуги для суспільства, консолідація інформації стає критичним завданням для забезпечення ефективної безпеки цієї галузі. Збільшення обсягів та складності інфраструктури, високий ризик виробничих процесів та загрози з боку тероризму та кібератак роблять актуальним об'єднання інформаційних ресурсів для вдосконалення систем безпеки.

Консолідація інформації для забезпечення безпеки хімічної інфраструктури – це процес збору, обробки та аналізу інформації про об'єкти хімічної інфраструктури, їхню діяльність та потенційні загрози. Ця інформація може бути використана для розробки та реалізації заходів щодо підвищення безпеки об'єктів хімічної інфраструктури.

Інформація, яка може бути використана для консолідації, включає:

– інформація про об'єкти хімічної інфраструктури, така як їхнє місцезнаходження, тип діяльності, використовувані хімічні речовини, особливості технологічних процесів тощо.

– інформація про потенційні загрози, такі як терористичні акти, кібератаки, аварії, природні катаклізми тощо.

– інформація про заходи безпеки, які вже впроваджені на об'єктах хімічної інфраструктури.

Консолідація інформації для забезпечення безпеки хімічної інфраструктури може здійснюватися різними способами. Одним із можливих підходів є створення єдиної інформаційної системи, яка буде об'єднувати інформацію з різних джерел. Іншим підходом є використання методів аналізу великих даних для обробки інформації з різних джерел та виявлення потенційних загроз.

Консолідація інформації для забезпечення кібербезпеки хімічної інфраструктури є важливим завданням, яке може сприяти підвищенню рівня захисту об'єктів хімічної інфраструктури від кібератак [16].

Інформація, яка може бути використана для консолідації в контексті кібербезпеки, включає:

– інформація про об'єкти хімічної інфраструктури, така як їхнє місцезнаходження, тип діяльності, використовувані хімічні речовини, особливості технологічних процесів, а також інфраструктура об'єкта, включаючи комп'ютерні системи, мережі та обладнання;

– інформація про потенційні загрози, такі як кібератаки, а також про заходи безпеки, які вже впроваджені на об'єктах хімічної інфраструктури для захисту від кібератак.

На першому етапі здійснюється збір інформації з різних джерел. Джерелами інформації можуть бути:

– об'єкти хімічної інфраструктури;
– уряди та державні органи;
– міжнародні організації;
– наукові установи;
– комерційні організації.

Інформація може бути зібрана за допомогою різних методів, включаючи:

– анкетування;
– інтерв'ю;
– аналіз відкритих джерел;
– збір інформації з інформаційних систем.

На другому етапі здійснюється оцінка зібраної інформації. Оцінка містить:

– відбір релевантної інформації;
– оцінку якості інформації;
– узгодження інформації з різних джерел,

На третьому етапі здійснюється обробка інформації. Обробка містить:

– форматування інформації;
– індексація інформації;
– архівування інформації

На четвертому етапі здійснюється зберігання інформації. Інформація повинна зберігатися в безпечному місці, з обмеженим доступом [18].

На п'ятому етапі здійснюється розповсюдження інформації. Інформація повинна бути доступна для всіх зацікавлених сторін, які відповідають за забезпечення кібербезпеки хімічної інфраструктури.

На шостому етапі здійснюється аналіз інформації. Аналіз містить:

– визначення потенційних кіберзагроз;

– розробку заходів кібербезпеки.

На сьомому етапі впроваджуються заходи кібербезпеки. Заходи кібербезпеки повинні бути спрямовані на зниження ризику кібератак на об'єкти хімічної інфраструктури.

Консолідація інформації для забезпечення кібербезпеки хімічної інфраструктури має ряд переваг, включаючи:

– покращення розуміння кіберзагроз;

Консолідована інформація може бути використана для покращення розуміння потенційних кіберзагроз, які можуть нести об'єкти хімічної інфраструктури. Це може сприяти розробці більш ефективних заходів кібербезпеки.

– покращення ефективності заходів кібербезпеки;

Консолідована інформація може бути використана для покращення ефективності заходів кібербезпеки, які вже впроваджені на об'єктах хімічної інфраструктури. Це може сприяти підвищенню рівня захисту об'єктів від кібератак.

– покращення взаємодії між різними зацікавленими сторонами.

Консолідована інформація може бути використана для покращення взаємодії між різними зацікавленими сторонами, які відповідають за забезпечення кібербезпеки хімічної інфраструктури. Це може сприяти більш ефективному реагуванню на потенційні кібератаки.

Консолідація інформації для забезпечення кібербезпеки хімічної інфраструктури є важливим завданням, яке може сприяти підвищенню рівня захисту об'єктів хімічної інфраструктури від кібератак.

1.4 Аналіз методів системного аналізу безпеки об’єктів

Системний аналіз безпеки об'єктів визначається як важлива галузь для вивчення, підвищення стійкості та захисту різноманітних об'єктів від потенційних

небезпек. Методи системного аналізу в цьому контексті є ключовим інструментом для розуміння, ідентифікації та управління ризиками, пов'язаними з безпекою об'єктів.

Системний аналіз безпеки об’єктів – це комплекс методів і засобів, що дозволяють оцінити стан безпеки об’єкта, виявити потенційні загрози та розробити заходи щодо їхнього усунення або зниження [19].

Мета системного аналізу безпеки об’єктів – забезпечити ефективний захист об’єкта від різноманітних видів загроз, включаючи:

– природні стихійні лиха (землетруси, повені, пожежі тощо);

– техногенні аварії (вибухи, пожежі, викиди токсичних речовин тощо);

– терористичні акти;

– кримінальні правопорушення (крадіжки, розбої, вбивства тощо);

– інші загрози, які можуть призвести до пошкодження або знищення об’єкта, загибелі людей або матеріальних збитків.

Системний аналіз безпеки об’єктів має такі основні етапи:

– опис об’єкта. На цьому етапі здійснюється всебічний аналіз об’єкта, включаючи його структуру, функціонування, технічні характеристики, а також зовнішні фактори, які можуть впливати на його безпеку;

– ідентифікація загроз. На цьому етапі проводиться аналіз потенційних загроз, які можуть виникнути для об’єкта. При цьому враховуються як внутрішні, так і зовнішні фактори;

– оцінка ризиків. На цьому етапі проводиться оцінка ймовірності настання загроз та їхнього впливу на об’єкт;

– розробка заходів щодо забезпечення безпеки. На цьому етапі розробляються заходи щодо усунення або зниження ризиків, пов’язаних із загрозами.

Для проведення системного аналізу безпеки об’єкта використовуються різні методи, які можна класифікувати за такими ознаками:

– за способом отримання інформації;

Аналітичний метод – передбачає аналіз інформації про об’єкт, загрози та ризики.

Опитувальний метод – передбачає проведення опитування експертів, які мають знання про об’єкт і загрози.

Експериментальний метод – передбачає проведення експериментів для оцінки ймовірності настання загроз та їхнього впливу на об’єкт.

– за ступенем деталізації;

Модельний метод – передбачає використання математичних моделей для оцінки ризиків.

Факторний метод – передбачає аналіз загроз та ризиків з урахуванням різних факторів.

Деталізований метод – передбачає проведення детального аналізу об’єкта, загроз та ризиків.

– за способом застосування.

Контрольний метод – передбачає використання методів для контролю стану безпеки об’єкта.

Прогнозний метод – передбачає використання методів для прогнозування стану безпеки об’єкта.

Розробницький метод – передбачає використання методів для розробки заходів щодо забезпечення безпеки об’єкта.

Найбільш поширеними методами системного аналізу безпеки об’єкта є:

– аналіз дерева відмов (FTA) - дозволяє визначити послідовність подій, які можуть призвести до відмови об’єкта;

– аналіз дерева подій (ETA) - дозволяє визначити послідовність подій, які можуть призвести до настання загрози;

– матричний аналіз ризиків (MAR) - дозволяє оцінити ризики, пов’язані із загрозами;

– аналіз критичних шляхів (CCP) - дозволяє визначити критичні елементи об’єкта, які є найбільш важливими для його безпеки.

Аналіз дерева відмов (FTA) - це метод, який дозволяє визначити послідовність подій, які можуть призвести до відмови об’єкта. FTA використовується для виявлення слабких місць у системі, які можуть призвести до її відмови.

FTA проводиться у кілька етапів:

– опис системи. На цьому етапі проводиться всебічний аналіз системи, включаючи її структуру, функціонування, технічні характеристики;

– ідентифікація відмов. На цьому етапі визначаються всі можливі відмови системи;

– побудова дерева відмов. На цьому етапі будується дерево відмов, яке відображає послідовність подій, які можуть призвести до кожної відмови;

– оцінка ризиків. На цьому етапі оцінюється ймовірність настання кожної відмови.

FTA є ефективним методом для виявлення слабких місць у системах. Однак, FTA може бути досить трудомістким методом, особливо для великих і складних систем.

Аналіз дерева подій (ETA) - це метод, який дозволяє визначити послідовність подій, які можуть призвести до настання загрози. ETA використовується для виявлення потенційних загроз і розробки заходів щодо їхнього усунення або зниження.

ETA проводиться у кілька етапів:

– опис системи. На цьому етапі проводиться всебічний аналіз системи, включаючи її структуру, функціонування, технічні характеристики;

– ідентифікація загроз. На цьому етапі визначаються всі можливі загрози, які можуть виникнути для системи;

– побудова дерева подій. На цьому етапі будується дерево подій, яке відображає послідовність подій, які можуть призвести до настання кожної загрози;

– оцінка ризиків. На цьому етапі оцінюється ймовірність настання кожної загрози.

ETA є ефективним методом для виявлення потенційних загроз. Однак, ETA може бути досить трудомістким методом, особливо для великих і складних систем.

Матричний аналіз ризиків (MAR) - це метод, який дозволяє оцінити ризики, пов’язані із загрозами. MAR використовується для кількісної оцінки ризиків, пов’язаних із загрозами.

MAR проводиться у кілька етапів:

– опис системи. На цьому етапі проводиться всебічний аналіз системи, включаючи її структуру, функціонування, технічні характеристики;

– ідентифікація загроз. На цьому етапі визначаються всі можливі загрози, які можуть виникнути для системи;

– оцінка ймовірності настання загроз. На цьому етапі оцінюється ймовірність настання кожної загрози;

– оцінка впливу загроз. На цьому етапі оцінюється вплив кожної загрози на систему;

– розрахунок ризиків. На цьому етапі розраховується ризик кожної загрози, як добуток ймовірності її настання та впливу.

MAR є ефективним методом для кількісної оцінки ризиків. Однак, MAR може бути досить складним методом, особливо для великих і складних систем.

Аналіз критичних шляхів (CCP) - це метод, який дозволяє визначити критичні елементи об’єкта, які є найбільш важливими для його безпеки. CCP використовується для розробки заходів щодо підвищення безпеки об’єкта [21].

CCP проводиться у кілька етапів:

– опис системи. На цьому етапі проводиться всебічний аналіз системи, включаючи її структуру, функціонування, технічні характеристики;

– побудова мережі взаємозв’язків між елементами системи. На цьому етапі будується мережа взаємозв’язків між елементами системи, яка відображає взаємодії між ними;

– розрахунок критичних шляхів. На цьому етапі розраховуються критичні шляхи в мережі взаємозв’язків, які є найбільш важливими для забезпечення безпеки системи.

ССР є ефективним методом для визначення критичних елементів об’єкта. Однак, ССР може бути досить складним методом, особливо для великих і складних систем.

З таблиці 1.1 видно, що всі розглянуті методи мають високу точність і об'єктивність. Практичність методів також висока, але вони вимагають певних ресурсів, тому їх економічність може бути різною.

Таблиця 1.1 – Порівняння методів системного аналізу безпеки об’єктів

Метод	Точність	Об'єктивність	Економічність	Швидкість проведення	Складність методу
Аналіз дерева відмов (FTA)	Висока	Висока	Середня	Висока	Висока
Аналіз дерева подій (ETA)	Висока	Висока	Середня	Висока	Висока
Матричний аналіз ризиків (MAR)	Середня	Середня	Висока	Середня	Середня
Аналіз критичних шляхів (ССР)	Висока	Висока	Висока	Висока	Висока

У порівнянні з іншими методами, аналіз дерева відмов (FTA) і аналіз дерева подій (ETA) мають високу швидкість проведення, але вони також є складними та

вимагають високого рівня знань і навичок. Матричний аналіз ризиків (MAR) є простішим і доступним методом, але він також має меншу точність і об'єктивність. Аналіз критичних шляхів (CCP) є складним і трудомістким методом, але він також є найбільш точним і об'єктивним.

У висновку можна зазначити, що системний аналіз є важливим інструментом для забезпечення безпеки об'єктів у сучасному середовищі, насиченому різноманітними загрозами та викликами. Методи системного аналізу безпеки об'єктів дозволяють структуровано та комплексно підходити до оцінки ризиків, ідентифікації вразливостей та розробки стратегій безпеки.

Усе вищезазначене свідчить про те, що системний аналіз безпеки об'єктів є необхідним етапом для вдосконалення та оптимізації заходів безпеки в умовах зростаючих загроз та змінюючись умов. Розуміння системних взаємозв'язків та комплексний підхід дають можливість вчасно реагувати на потенційні ризики та мінімізувати їхні негативні наслідки для безпеки об'єктів.

1.5 Аналіз сучасних методів забезпечення захисту інформаційного ресурсу

Швидкий технологічний прогрес та розвиток інтернет-технологій призвели до появи нових загроз та викликів для безпеки інформації. Саме тому аналіз та вдосконалення сучасних методів забезпечення захисту інформаційних ресурсів стає актуальним завданням у сфері кібербезпеки.

Одним з ключових напрямків захисту інформаційних ресурсів є кібербезпека, яка містить комплекс заходів з протидії кіберзагрозам та кібератакам. Методи кібербезпеки охоплюють широкий спектр заходів, включаючи шифрування даних, використання мережевих брандмауерів, систем виявлення вторгнень, антивірусного захисту та регулярне оновлення програмного забезпечення [22].

Сучасні методи забезпечення захисту інформаційного ресурсу можна класифікувати за різними ознаками.

За способом реалізації

– фізичні методи;

Забезпечують захист інформаційного ресурсу за допомогою фізичних засобів, таких як замки, охоронна сигналізація, контроль доступу тощо.

– логічні методи.

Забезпечують захист інформаційного ресурсу за допомогою логічних засобів, таких як системи аутентифікації, авторизації, шифрування тощо.

За сферою застосування

– методи захисту інформації на рівні користувача. Забезпечують захист інформації на рівні користувача, наприклад, від несанкціонованого доступу, використання або розголошення;

– методи захисту інформації на рівні мережі. Забезпечують захист інформації на рівні мережі, наприклад, від несанкціонованого доступу, розголошення або модифікації;

– методи захисту інформації на рівні сервера. Забезпечують захист інформації на рівні сервера, наприклад, від несанкціонованого доступу, використання, розголошення або знищення.

За ступенем автоматизації

– автоматизовані методи. Забезпечують захист інформації за допомогою автоматизованих засобів, таких як системи безпеки, антивірусні програми тощо;

– неавтоматизовані методи. Забезпечують захист інформації без використання автоматизованих засобів, наприклад, за допомогою людського фактора.

До основних сучасних методів забезпечення захисту інформаційного ресурсу відносяться:

– аутентифікація;

Це процес підтвердження особи користувача. Аутентифікація може здійснюватися за допомогою різних засобів, таких як паролі, смарт-карти, біометричні характеристики тощо.

– авторизація;

Це процес надання користувачеві права на доступ до ресурсу. Авторизація може здійснюватися на основі ролі користувача, його прав або на основі інших критеріїв.

– шифрування;

Це процес перетворення інформації в нечитабельний формат. Шифрування використовується для захисту інформації від несанкціонованого доступу.

– контроль доступу;

Це сукупність заходів, що забезпечують обмеження доступу до інформаційного ресурсу. Контроль доступу може здійснюватися на основі різних критеріїв, таких як роль користувача, його права, час, місце тощо.

– контроль вірусів;

Це комплекс заходів, що спрямовані на захист від комп'ютерних вірусів. Контроль вірусів може здійснюватися за допомогою антивірусних програм, фільтрів і інших засобів [24].

– системи виявлення вторгнень (IDS);

Це системи, що виявляють несанкціоновану активність в інформаційній системі. IDS можуть використовуватися для попередження про можливі атаки або для запобігання їм.

– системи запобігання вторгненням (IPS);

Це системи, що запобігають несанкціонованій активності в інформаційній системі. IPS можуть використовуватися для блокування атак або для їхнього відхилення.

Аутентифікація – це процес підтвердження особи користувача. Вона є першим кроком у забезпеченні безпеки інформаційного ресурсу, оскільки дозволяє відрізнити авторизованих користувачів від неавторизованих.

Аутентифікація може здійснюватися за допомогою різних засобів, таких як:

– паролі. Паролі є найпоширенішим засобом аутентифікації. Вони повинні бути складними та унікальними для кожного користувача;

– смарт-карти. Смарт-карти – це фізичні носії, які містять інформацію про користувача. Вони можуть використовуватися для аутентифікації в поєднанні з паролями або іншими засобами;

– біометричні характеристики. Біометричні характеристики – це унікальні фізичні або поведінкові характеристики людини, такі як відбитки пальців, голос, зір тощо. Вони можуть використовуватися для аутентифікації, оскільки є дуже складними для підроблення.

Авторизація – це процес надання користувачеві права на доступ до ресурсу. Після того, як користувач був успішно аутентифікований, йому необхідно надати право доступу до ресурсу.

Авторизація може здійснюватися на основі ролі користувача, його прав або на основі інших критеріїв. Наприклад, користувач може мати право доступу до певної інформації лише в тому випадку, якщо він є членом певної групи або має певні права [26].

Шифрування – це процес перетворення інформації в нечитабельний формат. Шифрування використовується для захисту інформації від несанкціонованого доступу.

Існує багато різних алгоритмів шифрування. Вибір алгоритму шифрування залежить від таких факторів, як рівень безпеки, необхідна швидкість шифрування та інші.

Контроль доступу – це сукупність заходів, що забезпечують обмеження доступу до інформаційного ресурсу. Контроль доступу може здійснюватися на основі різних критеріїв, таких як роль користувача, його права, час, місце тощо.

Контроль доступу може здійснюватися за допомогою таких засобів, як:

– системи аутентифікації та авторизації;

– механічні засоби контролю доступу, такі як замки, охоронна сигналізація тощо;

– логічні засоби контролю доступу, такі як політики безпеки, правила контролю доступу тощо.

Контроль вірусів – це комплекс заходів, що спрямовані на захист від комп'ютерних вірусів. Контроль вірусів може здійснюватися за допомогою антивірусних програм, фільтрів і інших засобів.

Антивірусні програми – це програмне забезпечення, яке використовується для виявлення і видалення вірусів. Антивірусні програми можуть використовуватися для сканування файлів, систем і мереж на наявність вірусів.

Фільтри – це програмні або апаратні засоби, які використовуються для блокування доступу до певних ресурсів або заборони певних дій. Фільтри можуть використовуватися для блокування доступу до вебсайту, файлів або програм, які є потенційно небезпечними.

Системи виявлення вторгнень (IDS) - це системи, що виявляють несанкціоновану активність в інформаційній системі. IDS можуть використовуватися для попередження про можливі атаки або для запобігання їм.

IDS працюють шляхом моніторингу мережевого трафіку та виявлення аномалій, які можуть бути ознаками атаки. IDS можуть генерувати сигнали тривоги, які можуть бути використані для попередження про можливі атаки або для їхнього запобігання.

Системи запобігання вторгненням (IPS) - це системи, що запобігають несанкціонованій активності в інформаційній системі. IPS можуть використовуватися для блокування атак або для їхнього відхилення.

IPS працюють шляхом моніторингу мережевого трафіку та блокування пакетів, які містять ознаки атак. IPS можуть бути ефективними для запобігання поширенню атак на інші системи.

Таблиця 1.2 – порівняння сучасних методів забезпечення захисту інформаційного ресурсу

Метод	Ефективність	Доступність	Економічність	Зручність використання
Аутентифікація	Висока	Висока	Середня	Середня
Авторизація	Висока	Висока	Середня	Середня

Продовження таблиці 1.2

Метод	Ефективність	Доступність	Економічність	Зручність використання
Шифрування	Висока	Середня	Вища	Складна
Контроль доступу	Висока	Висока	Середня	Середня
Контроль вірусів	Висока	Висока	Середня	Середня
Системи виявлення вторгнень (IDS)	Висока	Висока	Середня	Середня
Системи запобігання вторгненням (IPS)	Висока	Висока	Вища	Складна

З таблиці видно, що всі розглянуті методи забезпечення захисту інформаційного ресурсу є ефективними та доступними. Однак, вони відрізняються за рівнем економічності, зручністю використання та прогресивністю.

Аутентифікація, авторизація та контроль доступу є найбільш ефективними методами захисту інформаційного ресурсу. Вони забезпечують високий рівень захисту від несанкціонованого доступу, використання, розголошення, модифікації або знищення. Ці методи є також доступними та зручними в використанні.

Шифрування є також ефективним методом захисту інформаційного ресурсу. Воно забезпечує захист від несанкціонованого використання, розголошення та модифікації інформації. Однак, шифрування може бути складним у використанні та вимагає певних знань і навичок [27].

Системи виявлення вторгнень (IDS) та системи запобігання вторгненням (IPS) є найбільш прогресивними методами захисту інформаційного ресурсу. Вони використовують останні технології безпеки та дозволяють захистити інформаційний ресурс від сучасних загроз. Однак, ці методи можуть бути складними в налаштуванні та використанні.

Одним із важливих аспектів сучасних стратегій є комплексний підхід до кібербезпеки, охоплюючи технічні, організаційні та людські аспекти. Ефективні технічні заходи повинні поєднуватися з правильною організацією процесів управління доступом та підтримкою високого рівня свідомості серед персоналу.

Процес безпеки має бути динамічним та гнучким, оскільки кіберзагрози постійно еволюціонують. Регулярні аудити та оновлення стратегій захисту дозволяють підтримувати високий рівень захисту в обличчі нових викликів.

У цілому, враховуючи постійний прогрес технологій та розвиток кіберзагроз, важливо продовжувати вивчення та впровадження нових методів та технологій для забезпечення ефективного та надійного захисту інформаційних ресурсів. Тільки комплексний підхід та поєднання різних заходів можуть забезпечити стійкість інформаційних ресурсів у сучасному, динамічному кіберсередовищі.

1.6 Висновки та постановка задачі

Отже, у даному розділі був проведений аналіз сучасних стратегій забезпечення безпеки інформаційних ресурсів для хімічної інфраструктури, яка виступає критичною складовою інфраструктури регіону. Були розглянуті методи системного аналізу для оцінки безпеки об'єктів хімічної галузі.

У ході загального огляду теоретичного матеріалу були сформульовані наступні завдання:

– розробити захищений консолідований інформаційний ресурс, спрямований на забезпечення безпеки хімічної інфраструктури;

– розробити алгоритм оцінювання безпеки хімічних об'єктів;

– створити базу даних для зберігання інформації щодо безпеки хімічних систем;

– провести дослідження коректності та ефективності функціонування розробленого інформаційного ресурсу.

Здійснення всіх поставлених завдань сприятиме досягненню головної мети цієї роботи – розробки захищеного консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури регіону, що є важливою складовою для забезпечення стійкості та безпеки регіональних інфраструктурних об'єктів.

2 СТВОРЕННЯ ЗАХИЩЕНОГО КОНСОЛІДОВАНОГО ІНФОРМАЦІЙНОГО РЕСУРСУ АНАЛІЗУ БЕЗПЕКИ ІНФРАСТРУКТУРИ

У даному розділі буде здійснено аналіз особливостей створення інформаційного ресурсу аналізу безпеки хімічної інфраструктури, буде здійснена розробка бази даних та розроблено систему забезпечення захисту, після чого буде сформовано висновки по роботі виконаній в даному розділі.

2.1 Особливості створення інформаційного ресурсу аналізу безпеки хімічної інфраструктури

Хімічна інфраструктура, яка містить комплекс об'єктів і устаткування, є ключовою складовою промислових систем та відіграє значущу роль у глобальному економічному розвитку. Завдання забезпечення безпеки цієї складної інфраструктури вимагає ретельного аналізу та ефективних заходів, спрямованих на запобігання можливим ризикам та надзвичайним ситуаціям.

Створення інформаційного ресурсу для аналізу безпеки хімічної інфраструктури є актуальним завданням у зв'язку зі зростанням комплексності виробничих процесів та появою нових загроз. У цьому контексті важливо розглянути особливості створення такого інформаційного ресурсу, щоб забезпечити ефективний моніторинг, реагування на потенційні небезпеки та впровадження системи заходів превентивного захисту.

Хімічна інфраструктура є важливою складовою економіки та життєзабезпечення населення. Вона містить підприємства, що виробляють, зберігають та транспортують хімічні речовини. Безпека хімічної інфраструктури є важливою державною проблемою, оскільки її порушення може призвести до серйозних наслідків для населення, економіки та екології.

Для забезпечення безпеки хімічної інфраструктури необхідно мати ефективну систему управління безпекою. Така система повинна містити комплекс заходів, що спрямовані на виявлення, оцінку та зниження ризиків безпеки.

Одним з елементів системи управління безпекою хімічної інфраструктури є інформаційний ресурс аналізу безпеки. Цей ресурс повинен містити інформацію про об'єкти хімічної інфраструктури, їхні особливості, потенційні загрози та заходи безпеки.

Створення інформаційного ресурсу аналізу безпеки хімічної інфраструктури має ряд особливостей, які необхідно враховувати.

Перш за все, такий ресурс повинен бути комплексним. Він повинен містити інформацію про всі об'єкти хімічної інфраструктури, які знаходяться в зоні відповідальності певної організації.

По-друге, ресурс повинен бути актуальним. Інформація, що міститься в ньому, повинна регулярно оновлюватися з урахуванням змін, що відбуваються в об'єктах хімічної інфраструктури та в середовищі їхнього функціонування.

По-третє, ресурс повинен бути уніфікованим. Інформація, що міститься в ньому, повинна бути представлена в єдиному форматі, що дозволить її легко обробляти та аналізувати [30].

По-четверте, ресурс повинен бути зручним у використанні. Користувачі повинні мати можливість швидко і легко знаходити необхідну інформацію.

Хімічна інфраструктура є важливим елементом економіки та життєзабезпечення населення. Вона містить підприємства, що виробляють, зберігають та транспортують хімічні речовини. Безпека хімічної інфраструктури є важливою державною проблемою, оскільки її порушення може призвести до серйозних наслідків для населення, економіки та екології.

Розробка інформаційного ресурсу для аналізу безпеки хімічної інфраструктури є складним і відповідальним завданням. Цей ресурс повинен забезпечити ефективний контроль безпеки хімічних об'єктів, враховуючи їхню специфіку та потенційні ризики.

Основні аспекти створення інформаційного ресурсу для аналізу безпеки хімічної інфраструктури:

– специфіка об'єктів хімічної інфраструктури: Хімічні підприємства містять різноманітні об'єкти, які можуть бути підвищеної небезпеки. Створення інформаційного ресурсу повинно враховувати специфіку кожного об'єкта та його можливі ризики;

– аналіз технічних процесів: Інформаційний ресурс повинен охоплювати детальний аналіз технічних процесів, що відбуваються на об'єктах хімічної інфраструктури. Це містить вивчення хімічних реакцій, технологічних параметрів та систем контролю;

– збір та обробка даних про безпеку: Розробка системи збору та обробки даних про безпеку, яка містить інформацію про виробничі параметри, рівень токсичності, температурні умови, тиск, витрати реагентів, а також моніторинг захисту промислових об'єктів;

– кібербезпека та захист інформації: Врахування особливостей кібербезпеки для хімічних об'єктів, оскільки порушення цілісності чи конфіденційності інформації може призвести до серйозних наслідків. Розробка заходів для захисту від кібератак та несанкціонованого доступу до систем;

– аналіз можливих небезпек: Оцінка можливих загроз та розробка системи аналізу ризиків, яка враховує можливі небезпеки для здоров'я людей, довкілля та економічних втрат у випадку аварій;

– моніторинг та реагування на аварії: Створення системи моніторингу, яка надає можливість виявлення можливих аварій чи надзвичайних ситуацій, а також розробка ефективних механізмів реагування та надання швидкої інформації для зменшення наслідків;

– нормативне регулювання: Врахування вимог нормативно-правового поля, яке регулює діяльність хімічних підприємств, та впровадження системи відповідності до стандартів та норм безпеки.

Створення інформаційного ресурсу для аналізу безпеки хімічної інфраструктури – завдання високої важливості, яке вимагає комплексного підходу та урахування унікальних характеристик цієї промислової галузі. Розгляд

особливостей впровадження такого ресурсу підкреслив необхідність детального аналізу технічних процесів, систем безпеки, кіберзахисту та відповідності нормативам.

Створення ефективної системи аналізу безпеки в хімічній інфраструктурі передбачає глибоке вивчення та управління можливими ризиками, а також оперативне реагування на надзвичайні ситуації. Забезпечення конфіденційності та цілісності інформації стає критичним для успішної реалізації системи моніторингу та контролю в цій чутливій галузі.

В результаті, створення інформаційного ресурсу для аналізу безпеки хімічної інфраструктури є стратегічно важливою ініціативою, яка сприятиме запобіганню можливим аваріям та загрозам, забезпечуючи стійкість та безпеку промислових об'єктів.

2.2 Розробка БД консолідованого інформаційного ресурсу аналізу безпеки хімічної інфраструктури методом сутність-зв'язок.

На етапі визначення вимог для розробки бази даних консолідованого інформаційного ресурсу аналізу безпеки хімічної інфраструктури важливо провести аналіз потреб у даних. Це охоплює ідентифікацію суб'єктів даних, визначення типів інформації та аналіз потреб різних категорій користувачів, від пацієнтів до адміністраторів. Ключовою задачею є визначення основних функціональних вимог до системи, зокрема зберігання, оновлення, вилучення даних та здійснення аналізу та звітності.

Процес встановлення умов доступу включає уточнення, як різні користувачі можуть отримати доступ до різних категорій інформації, враховуючи їхні ролі та повноваження. Також визначаються механізми захисту, такі як шифрування та контроль доступу, для забезпечення безпеки та конфіденційності даних.

Важливим етапом є адаптація до вимог сучасних стандартів та законодавства. Визначаються вимоги до збереження та обробки даних відповідно до вимог законодавства, а також враховуються стандарти безпеки, забезпечуючи відповідність з ISO/IEC 27001.

Такий підхід до визначення вимог передбачає тісну співпрацю з усіма сторонами, щоб забезпечити, що розроблювана база даних буде відповідати високим стандартам безпеки та відповідати потребам користувачів в контексті аналізу безпеки хімічної інфраструктури [31].

Проєктування схеми бази даних (БД) для консолідованого інформаційного ресурсу аналізу безпеки хімічної інфраструктури є етапом, який передбачає створення структурованої моделі для ефективного зберігання та організації даних.

Першим кроком в методі проєктування бази даних сутність-зв'язок є визначення сутностей, які повністю відображатимуть інформаційні потреби користувача майбутньої бази даних.

Виділимо потрібні сутності:

- КОРИСТУВАЧІ
- РЕЗУЛЬТАТИ
- ХІМІЧНА ІНФРАСТРУКТУРА
- ОБ’ЄКТИ
- СИСТЕМИ ЗАХИСТУ ОБ’ЄКТА

Наступним кроком визначимо зв’язки між обраними сутностями:

- КОРИСТУВАЧ виконує РЕЗУЛЬТАТИ
- РЕЗУЛЬТАТИ містять ХІМІЧНА ІНФРАСТРУКТУРА
- ХІМІЧНА ІНФРАСТРУКТУРА містить ОБ’ЄКТИ
- ОБ’ЄКТИ мають СИСТЕМИ ЗАХИСТУ ОБ’ЄКТА

Далі визначаємо атрибути обраних сутностей та ключі для кожної із сутностей:

– Сутність «КОРИСТУВАЧІ», яка буде містити наступні атрибути <Код користувача>, Ім’я, Прізвище, Логін, Пароль, Роль;

– Сутність «РЕЗУЛЬТАТИ», яка буде містити наступні атрибути <Код аналізу захисту>, Код хімічної інфраструктури, Код, користувача, Дата аналізу, Висновок, Додаткові деталі;

– Сутність «ХІМІЧНА ІНФРАСТРУКТУРА», яка буде містити наступні атрибути <Код хімічної інфраструктури>, Назва інфраструктури, Опис інфраструктури

– Сутність «ОБ’ЄКТИ», яка буде містити наступні атрибути <Код об’єкта інфраструктури >, Код хімічної інфраструктури, Назва об’єкта інфраструктури, Тип об’єкта інфраструктури, Опис об’єкта інфраструктури;

– Сутність «СИСТЕМИ ЗАХИСТУ ОБ’ЄКТА», яка буде містити наступні атрибути <Код системи захисту об’єкта >, Код об’єкта інфраструктури, Назва системи захисту об’єкта, Тип системи захисту об’єкта, Опис системи захисту об’єкта;

Далі необхідно визначити ступені зв’язків між сутностями.

Визначення ступеня зв`язку та класу належності сутностей «КОРИСТУВАЧІ» та «РЕЗУЛЬТАТИ» (рис. 2.1).

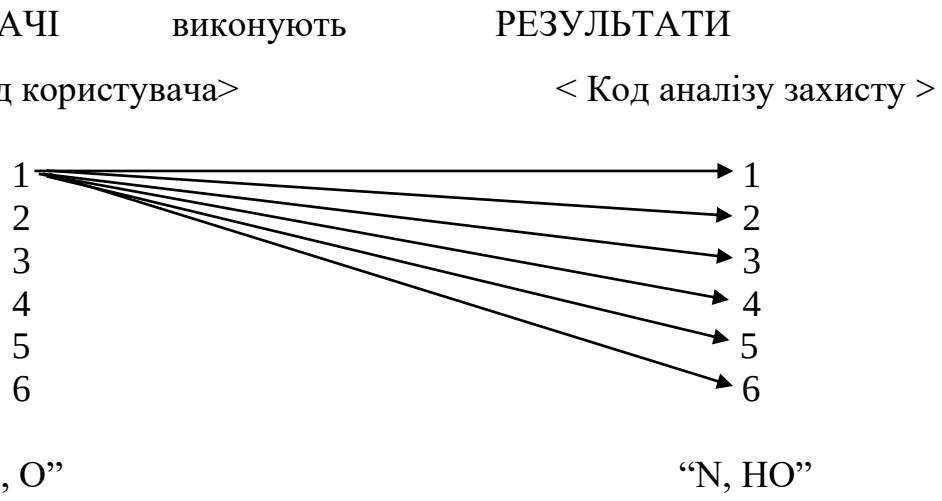


Рисунок 2.1 – Аналіз зв’язку сутностей «КОРИСТУВАЧІ» та «РЕЗУЛЬТАТИ»

Визначення ступеня зв`язку та класу належності сутностей «РЕЗУЛЬТАТИ» та «ХІМІЧНА ІНФРАСТРУКТУРА» (рис. 2.2).

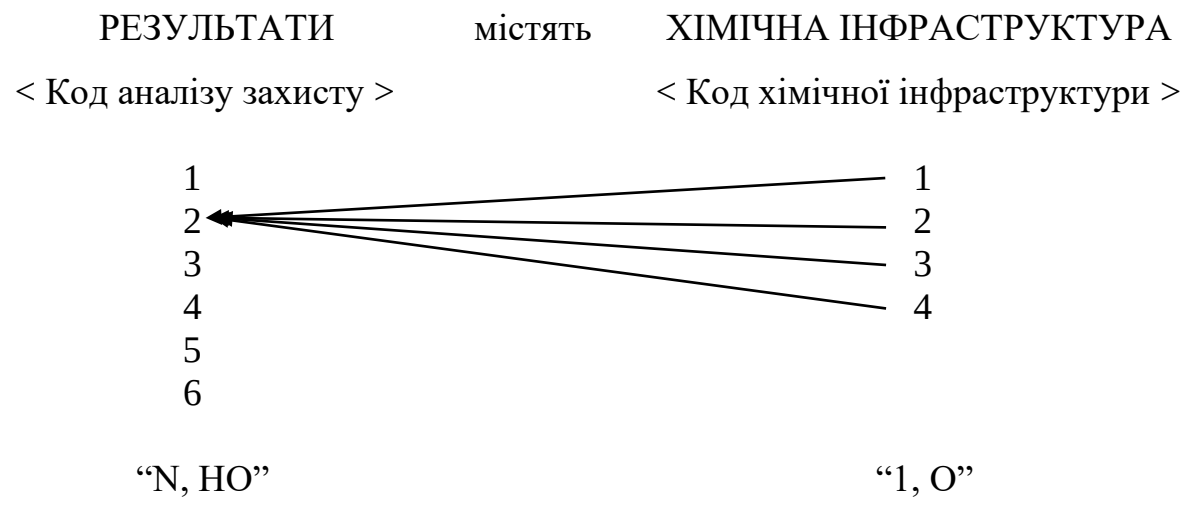


Рисунок 2.2 – Аналіз зв’язку сутностей «РЕЗУЛЬТАТИ» та «ХІМІЧНА ІНФРАСТРУКТУРА»

Визначення ступеня зв`язку та класу належності сутностей «ХІМІЧНА ІНФРАСТРУКТУРА» та «ОБ’ЄКТИ» (рис. 2.3).

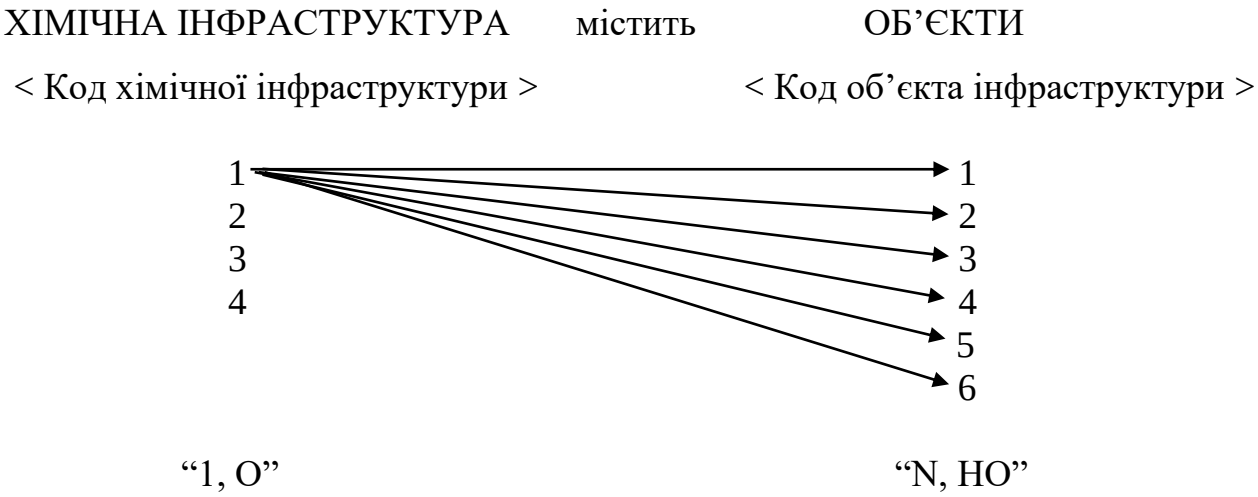


Рисунок 2.3 – Аналіз зв’язку сутностей «ХІМІЧНА ІНФРАСТРУКТУРА» та «ОБ’ЄКТИ»

Визначення ступеня зв`язку та класу належності сутностей «ОБ’ЄКТИ» та «СИСТЕМИ ЗАХИСТУ ОБ’ЄКТА» (рис. 2.4).

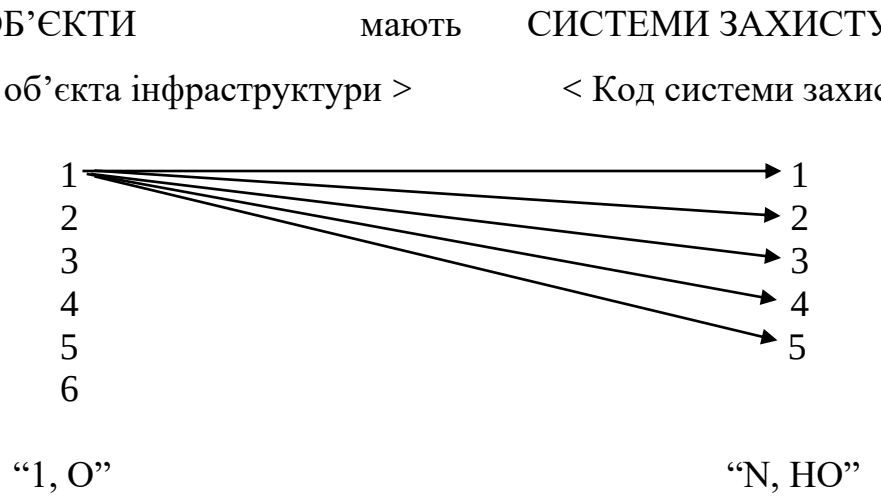


Рисунок 2.4 – Аналіз зв’язку сутностей «ОБ’ЄКТИ» та «СИСТЕМИ ЗАХИСТУ ОБ’ЄКТА

Логічна схема визначає структуру даних на рівні концепцій та відображає спосіб, яким дані пов'язані між собою. Вона є основою для подальших етапів проєктування та визначає загальний ландшафт бази даних. Важливо враховувати потреби користувачів та бізнес-вимоги при розробці логічної схеми для забезпечення відповідності функціональності та ефективності системи аналізу безпеки хімічної інфраструктури.

Розроблена база даних спроєктована для зручного аналізу безпеки відзначається ієрархічною структурою, що відображає взаємозв'язки між різними аспектами безпеки, такими як користувачі, інфраструктура, об'єкти, системи захисту, результати аналізу та інциденти безпеки. Використання зовнішніх ключів сприяє створенню зв'язків між таблицями, що спрощує взаємодію з даними та гарантує їх цілісність. Таблиця "Результати Аналізу" дозволяє ефективно зберігати та відстежувати результати проведених аналізів безпеки для різних об'єктів та систем захисту. Таблиця "Інциденти Безпеки" надає можливість відстежування історії інцидентів та їх наслідків, сприяючи виявленню шаблонів та покращенню стратегій безпеки. Зв'язок між об'єктами, системами захисту та інцидентами забезпечує можливість аналізу залишкових ризиків та оцінки ефективності застосованих заходів безпеки. Легкий доступ до інформації, використання індексів та оптимізована структура забезпечують швидкий та ефективний аналіз. Запис інформації про результати аналізу та інциденти сприяє

автоматизованому створенню звітів та аналізу трендів. Загальна систематизація та централізація інформації у базі даних дозволяють організації ефективно виявляти, реагувати та вдосконалювати свої стратегії безпеки.

2.3 Розробка ER-моделі майбутньої бази даних

В результаті розробки була створена ER-моделі майбутньої бази даних зображена на рисунку 2.5.

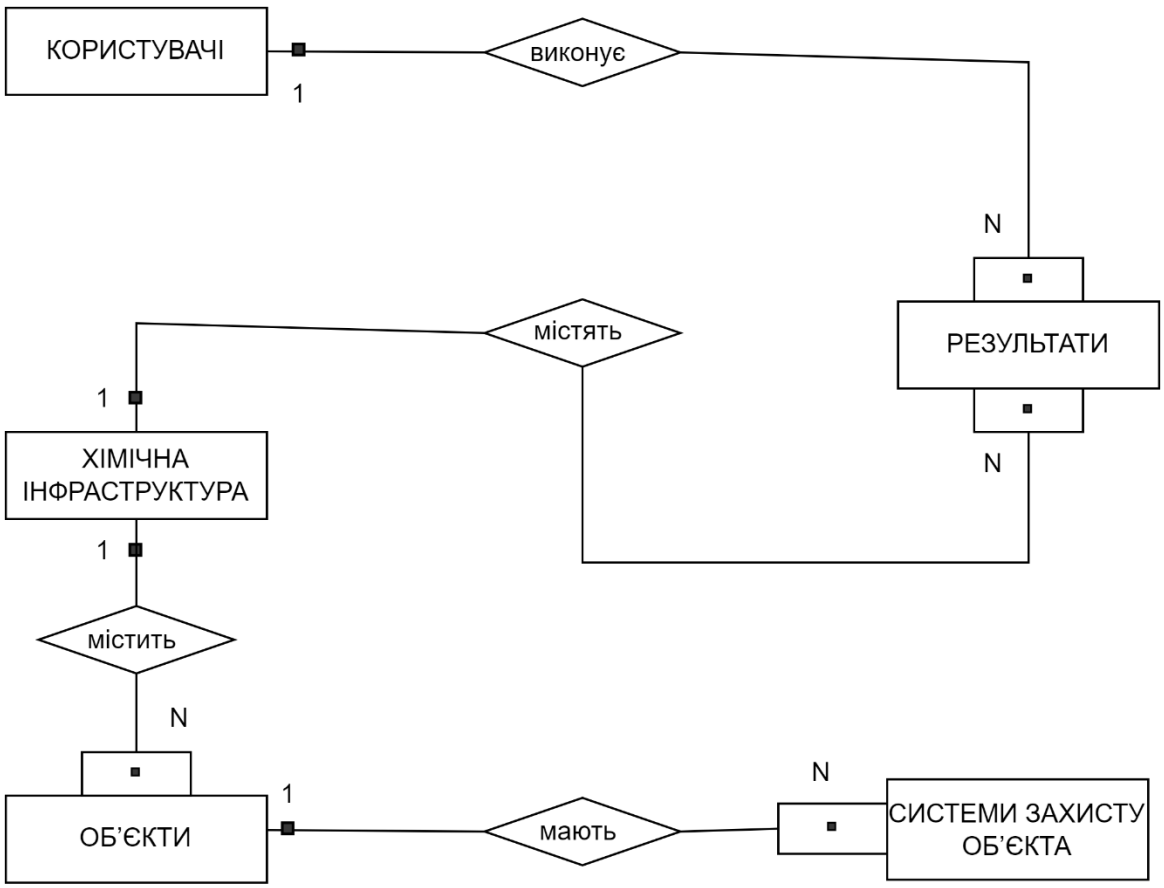


Рисунок 2.5 – ER-моделі майбутньої бази даних

Також були отриманні наступні попередні відношення та атрибути:

– Сутність «КОРИСТУВАЧІ», яка буде містити наступні атрибути <Код користувача>, Ім'я, Прізвище, Логін, Пароль, Роль;

– Сутність «РЕЗУЛЬТАТИ», яка буде містити наступні атрибути <Код аналізу захисту>, Код хімічної інфраструктури, Код, користувача, Дата аналізу, Висновок, Додаткові деталі;

– Сутність «ХІМІЧНА ІНФРАСТРУКТУРА», яка буде містити наступні атрибути <Код хімічної інфраструктури>, Назва інфраструктури, Опис інфраструктури

– Сутність «ОБ'ЄКТИ», яка буде містити наступні атрибути <Код об'єкта інфраструктури >, Код хімічної інфраструктури, Назва об'єкта інфраструктури, Тип об'єкта інфраструктури, Опис об'єкта інфраструктури;

– Сутність «СИСТЕМИ ЗАХИСТУ ОБ'ЄКТА», яка буде містити наступні атрибути <Код системи захисту об'єкта >, Код об'єкта інфраструктури, Назва системи захисту об'єкта, Тип системи захисту об'єкта, Опис системи захисту об'єкта;

2.4 Отримання кінцевих відношень БД за методом нормалізації відношень.

Під час створення реляційної бази даних необхідно розробити оптимальну структуру, яка включатиме певну кількість та тип атрибутів для однієї або кількох таблиць. При цьому важливо, щоб сукупність атрибутів була такою, що мінімізує дублювання даних та спрощує процедури обробки та оновлення. Для досягнення цієї мети використовується апарат нормалізації початкових відношень, який дозволяє привести будь-яку початкову таблицю до першої, другої чи третьої форми.

Здійснимо визначення інформації, необхідної для роботи захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону:

– Сутність «КОРИСТУВАЧІ», яка буде містити наступні атрибути <Код користувача>, Ім'я, Прізвище, Логін, Пароль, Роль;

– Сутність «РЕЗУЛЬТАТИ», яка буде містити наступні атрибути <Код аналізу захисту>, Код хімічної інфраструктури, Код, користувача, Дата аналізу, Висновок, Додаткові деталі;

– Сутність «ХІМІЧНА ІНФРАСТРУКТУРА», яка буде містити наступні атрибути <Код хімічної інфраструктури>, Назва інфраструктури, Опис інфраструктури

– Сутність «ОБ'ЄКТИ», яка буде містити наступні атрибути <Код об'єкта інфраструктури >, Код хімічної інфраструктури, Назва об'єкта інфраструктури, Тип об'єкта інфраструктури, Опис об'єкта інфраструктури;

– Сутність «СИСТЕМИ ЗАХИСТУ ОБ'ЄКТА», яка буде містити наступні атрибути <Код системи захисту об'єкта >, Код об'єкта інфраструктури, Назва системи захисту об'єкта, Тип системи захисту об'єкта, Опис системи захисту об'єкта;

Для реалізації поставленої задачі універсальне відношення буде мати вигляд:

R(Код користувача, Ім'я, Прізвище, Логін, Пароль, Роль, Код аналізу захисту, Код хімічної інфраструктури, Дата аналізу, Висновок, Додаткові деталі, Назва інфраструктури, Опис інфраструктури, Код об'єкта інфраструктури Назва об'єкта інфраструктури, Тип об'єкта інфраструктури, Опис об'єкта інфраструктури, Код системи захисту об'єкта,Назва системи захисту об'єкта, Тип системи захисту об'єкта, Опис системи захисту об'єкта).

Первинний ключ < Код користувача>

Проведемо нормалізацію відношень за трьома формами.

1НФ:

R1(Код користувача, Ім'я, Прізвище, Логін, Пароль, Роль, Код аналізу захисту, Код хімічної інфраструктури, Дата аналізу, Висновок, Додаткові деталі, Назва інфраструктури, Опис інфраструктури, Код об'єкта інфраструктури Назва об'єкта інфраструктури, Тип об'єкта інфраструктури, Опис об'єкта інфраструктури, Код системи захисту об'єкта,Назва системи захисту об'єкта, Тип системи захисту об'єкта, Опис системи захисту об'єкта).

2НФ:

R2(<Код аналізу захисту>, Код хімічної інфраструктури, Дата аналізу, Висновок, Додаткові деталі, Назва інфраструктури, Опис інфраструктури, Код об’єкта інфраструктури Назва об’єкта інфраструктури, Тип об’єкта інфраструктури, Опис об’єкта інфраструктури, Код системи захисту об’єкта,Назва системи захисту об’єкта, Тип системи захисту об’єкта, Опис системи захисту об’єкта).

R3(<Код користувача>, Ім’я, Прізвище, Логін, Пароль, Роль)

3 НФ:

R4 (<Код користувача>, Ім’я, Прізвище, Логін, Пароль, Роль);

R5 (<Код аналізу захисту>, Код хімічної інфраструктури, Код, користувача, Дата аналізу, Висновок, Додаткові деталі);

R6 (<Код хімічної інфраструктури>, Назва інфраструктури, Опис інфраструктури)

R7 (<Код об’єкта інфраструктури >, Код хімічної інфраструктури, Назва об’єкта інфраструктури, Тип об’єкта інфраструктури, Опис об’єкта інфраструктури);

R8 (<Код системи захисту об’єкта >, Код об’єкта інфраструктури, Назва системи захисту об’єкта, Тип системи захисту об’єкта, Опис системи захисту об’єкта);

Виходячи з проєктування, кінцевими відношеннями будуть: R4, R5, R6, R7, R8. На їх основі й відбуватиметься розробка бази даних захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону.

2.5 Проектування звітів.

Створення звітів є важливою складовою консолідованого інформаційного ресурсу, який призначений для дослідження хімічної інфраструктури та спрощення аналізу даних для профспілкових організацій. Розробка ефективних звітів передбачає представлення агрегованої статистичної інформації у зручному

форматі, спрямованому на подальший аналіз. У даному випадку пропонується створення ряду звітів, які відобразять ключові показники діяльності у галузі хімічної інфраструктури та сприятимуть обґрунтованим висновкам.

Звіт: Список користувачів (UserList).

Атрибути:

– код користувача (UserID) - унікальний ідентифікатор користувача;
– ім’я (FirstName) - ім'я користувача;
– прізвище (LastName) - прізвище користувача;
– логін (Login) - логін користувача для входу в систему;
– роль (Role) -роль, яку виконує користувач в системі.

Звіт: Список результатів (AnalysisResultList).

Атрибути:

– код аналізу захисту (AnalysisCode) - унікальний ідентифікатор аналізу захисту;
– код хімічної інфраструктури (ChemicalInfrastructureCode) - унікальний ідентифікатор хімічної інфраструктури;
– код користувача (UserID) - унікальний ідентифікатор користувача, який провів аналіз;
– дата аналізу (AnalysisDate) - дата проведення аналізу;
– висновок (Conclusion) - висновок результатів аналізу;
– додаткові деталі (AdditionalDetails) - додаткова інформація або деталі аналізу.

Звіт: Список хімічних інфраструктур (ChemicalInfrastructureList).

Атрибути:

– код хімічної інфраструктури (ChemicalInfrastructureCode) - унікальний ідентифікатор хімічної інфраструктури;
– назва інфраструктури (InfrastructureName) - назва хімічної інфраструктури;
– опис інфраструктури (InfrastructureDescription) - опис хімічної інфраструктури.

Звіт: Список об'єктів інфраструктури (InfrastructureObjectList).

Атрибути:

– код об’єкта інфраструктури (ObjectCode) - унікальний ідентифікатор об'єкта інфраструктури;

– код хімічної інфраструктури (ChemicalInfrastructureCode) - унікальний ідентифікатор хімічної інфраструктури;

– назва об’єкта інфраструктури (ObjectName) - назва об'єкта інфраструктури;

– тип об’єкта інфраструктури (ObjectType) - тип об'єкта інфраструктури;

– опис об’єкта інфраструктури (ObjectDescription) - опис об'єкта інфраструктури.

Звіт: Список системи захисту об'єкта (SecuritySystemList).

Атрибути:

– код системи захисту об’єкта (SecuritySystemCode) - унікальний ідентифікатор системи захисту об'єкта;

– код об’єкта інфраструктури (ObjectCode) - унікальний ідентифікатор об'єкта інфраструктури, до якого відноситься система захисту;

– назва системи захисту об’єкта (SystemName) - назва системи захисту об'єкта;

– тип системи захисту об’єкта (SystemType) - тип системи захисту об'єкта;

– опис системи захисту об’єкта (SystemDescription) - опис системи захисту об'єкта.

Звіт: Типи захисту хімічної інфраструктури.

Атрибути:

– назва об'єкту інфраструктури (ObjectName) - назва об'єкта хімічної інфраструктури;

– тип захисту (ProtectionType) - тип захисту, який застосовується для даного об'єкта хімічної інфраструктури.

Звіт: Типи об'єктів хімічної інфраструктури.

Атрибути:

– назва інфраструктури (InfrastructureName) - назва хімічної інфраструктури;

– тип об'єкта (ObjectType) - тип об'єкта хімічної інфраструктури.

Звіт: Рейтинг об'єктів хімічної інфраструктури за кількістю засобів захисту.

Атрибути:

– назва об'єкта інфраструктури (ObjectName) - назва об'єкта хімічної інфраструктури;

– кількість засобів захисту (SecurityDeviceCount) - кількість засобів захисту, які застосовуються для даного об'єкта хімічної інфраструктури.

2.6 Забезпечення захисту створеного консолідованого інформаційного ресурсу.

В сучасних умовах захист інформаційних ресурсів комплексного підходу, зокрема використання фаєрволів як важливого елементу системи безпеки. Фаєрволи відіграють ключову роль у керуванні мережевим трафіком, фільтрації небезпечного та несанкціонованого трафіку. Однак, для забезпечення ефективного захисту, необхідно враховувати різні аспекти цього процесу.

Спроєктований фаєрвол повинен враховувати чіткі правила керування трафіком, визначати правила доступу для конкретних IP-адрес, портів та протоколів. Окрім того, важливо використовувати аналіз заголовків та пакетів для ідентифікації типів та джерел трафіку, а також виявлення підозрілої активності.

Фаєрволи повинні мати можливість обмежувати доступ до зовнішніх ресурсів та Інтернету для контролю ризиків безпеки. Важливо використовувати фільтрацію URL для блокування доступу до сайтів із сумнівною або небажаною активністю.

При розробці захисту від SQL-ін'єкцій важливо використовувати параметризовані запити для уникнення введення SQL-коду, що може призвести до небажаних операцій або розголошення конфіденційної інформації. Це можна досягти через валідацію та екранування введених даних перед використанням їх у SQL-запитах. Використання параметризованих збережених процедур також може зменшити ризик SQL-ін'єкцій, оскільки вони можуть бути підготовлені та скомпільовані заздалегідь. Крім того, важливо обмежувати привілеї користувачів,

надаючи їм лише необхідні для завдань привілеї, тим самим зменшуючи можливість використання SQL-ін'єкцій для несанкціонованого доступу. Нарешті, використання механізмів фаєрволів та інших засобів безпеки, таких як інспекція введених даних та моніторинг бази даних, може допомогти вчасно виявляти та запобігати атакам SQL-ін'єкцій. Важливо застосовувати комплексний підхід для максимального рівня захисту.

Отже, розробимо алгоритм захисту, який буде забезпечувати роботу фаєрволу та захищати розроблений інформаційний ресурс від SQL-ін'єкцій (рис. 2.5).

Розглянемо більш детально розроблений алгоритм крок за кроком:

Крок 1. Початок

Початок алгоритму захисту.

Крок 2. Очікування запиту на сервер.

Сервер очікує отримання запиту.

Крок 3. Отримання запиту на сервер.

Сервер отримує запит.

Крок 4. Отримання протоколу запиту

Отримання типу протоколу за яким надійшов запит.

Крок 5. Запит здійснюється по протоколу HTTPS?

Перевірка чи здійснюється запит за протоколом HTTPS, дана перевірка здійсненна для забезпечення передавання даних лише по захищеному протоколу та для забезпечення обробки запитів лише за цим протоколом.

Якщо ТАК: Крок 5.1. Чи дозволений даний тип запиту? Якщо ТАК: Крок 6, якщо НІ: Крок 5.2.

Якщо НІ: Крок 5.2 Відхилення запиту, та перехід до кроку 17.

Відхилення запиту до сервера.

Крок 6. Чи дозволений даний тип запиту?

Перевірка чи дозволений даний тип запитів до сервера.

Якщо НІ: Крок 5.2.

Якщо ТАК: Крок 7.

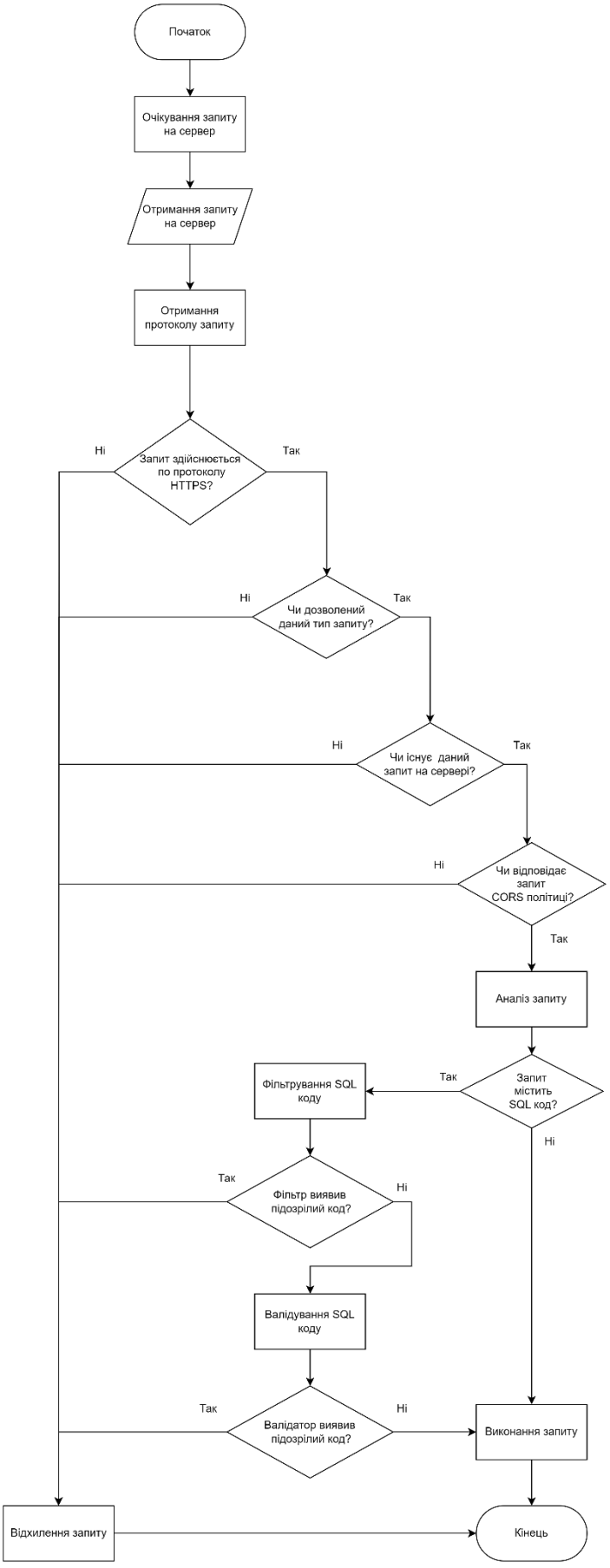


Рисунок 2.6 – Алгоритм розробленого захисту

Крок 7. Чи існує даний запит на сервері?

Перевірка чи взагалі існує даний запит на сервері.

Якщо НІ: Крок 5.2.

Якщо ТАК: Крок 8.

Крок 8. Чи відповідає запит CORS політиці?

Перевірка того, чи відповідає запит політиці обміну ресурсами між доменами (CORS).

Якщо НІ: Крок 5.2.

Якщо ТАК: Крок 9.

Крок 9. Аналіз запиту

Початок аналізу запиту

Крок 10. Запит містить SQL код?

Якщо НІ: Крок 16.

Якщо ТАК: Крок 11.

Крок 11. Фільтрування SQL коду

Фільтрація SQL коду з перевіркою на шкідливі елементи для запобігання ін'єкції.

Крок 12. Фільтр виявив підозрілий код?

Перевірка чи фільтр виявив підозрілий SQL код. Фільтрація SQL-коду може виявити підозрілі конструкції або некоректний синтаксис.

Якщо ТАК: Крок 5.2.

Якщо НІ: Крок 13.

Крок 13. Валідування SQL коду

Проведення валідації на підозрілі символи в SQL коді, щоб уникнути використання його для ін'єкцій та інших безпекових загроз.

Крок 14. Валідатор виявив підозрілий код?

Перевірка чи валідатор виявив підозрілі символи в SQL коді

Якщо ТАК: Крок 5.2.

Якщо НІ: Крок 16.

Крок 16. Виконання запиту.

Виконання безпечного запиту сервером.

Крок 17. Кінець

Кінець алгоритму.

Отже, можна зробити висновок, що розроблений алгоритм захисту на основі фаєрволу та захисту від SQL-ін'єкцій для консолідованого інформаційного ресурсу є досить надійним і відповідає сучасним вимогам, що забезпечить захист даних від витоку та забезпечить їх цілісність.

2.7 Висновки до розділу.

Отже, у даному розділі був розроблений захищений консолідований інформаційний ресурс для аналізу безпеки хімічної інфраструктури в регіоні. Однією з ключових вимог при розробці було врахування специфіки хімічної інфраструктури даного регіону, що визначило необхідність створення адаптивної та ефективної системи.

У процесі роботи була розроблена база даних, спрямована на консолідацію та аналіз різноманітних даних з безпеки хімічних об'єктів. Ця база даних забезпечує зручний доступ до інформації, її зберігання та подальший аналіз, що є ключовим для вдосконалення систем безпеки та реагування на можливі небезпеки.

Додатково, було проведено проектування звітів, спрямованих на зручне та ефективне представлення та аналіз отриманих даних. Це дозволяє користувачам отримувати необхідну інформацію в найбільш зрозумілій та зручній формі, сприяючи прийняттю обґрунтованих рішень щодо безпеки хімічних об'єктів.

Особлива увага була приділена аспектам забезпечення захисту інформації. Враховуючи чутливість даних про безпеку, були впроваджені ефективні заходи для забезпечення конфіденційності та цілісності інформації.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ЗАХИЩЕНОГО КОНСОЛІДОВАНОГО ІНФОРМАЦІЙНОГО РЕСУРСУ ТА СИСТЕМНИЙ АНАЛІЗ БЕЗПЕКИ ХІМІЧНОЇ ІНФРАСТРУКТУРИ

При виконанні даної роботи буде здійснено обґрунтування вибору мови програмування та обґрунтування вибору системи управління базою даних, після чого буде здійснена практична реалізація бази даних та модулів захисту, після чого буде реалізована система оцінювання захисту інфраструктури та сформовано висновки.

3.1 Обґрунтування вибору СУБД

Хімічна інфраструктура є важливим елементом економіки та безпеки будь-якої країни. Для забезпечення її безпеки необхідно мати ефективну систему моніторингу та аналізу стану об'єктів хімічної інфраструктури. Ця система повинна бути здатною збирати, зберігати та аналізувати великі обсяги різноманітних даних, зокрема, дані про стан обладнання, технологічних процесів, навколишнього середовища та т.д.

Для розробки такої системи необхідно вибрати відповідну систему управління базами даних (СУБД). СУБД повинна відповідати таким вимогам:

- здатність зберігати та обробляти великі обсяги даних;
- надійність та безпека;
- можливість масштабування.

На ринку існує безліч СУБД, які можуть бути використані для розробки системи системного аналізу безпеки хімічної інфраструктури. Серед них можна виділити такі:

- Oracle;

Одна з найпопулярніших СУБД у світі. Має широкий спектр функцій та можливостей, але є досить дорогою.

- Microsoft SQL Server;

Ще одна популярна СУБД. Має схожі з Oracle можливості, але є більш доступною.

– MySQL;

Відкрита СУБД, яка є популярною серед розробників. Має хороші характеристики продуктивності та масштабування, але потребує додаткової настройки для забезпечення безпеки.

– PostgreSQL.

Відкрита СУБД, яка має широкий спектр функцій та можливостей. Має хороші характеристики продуктивності, масштабування та безпеки.

На основі проведеного аналізу можна зробити висновок, що Postgres є найкращим вибором для розробки захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури.

Postgres підтримує зберігання даних у масштабі терабайтів і петабайтів. Це дозволяє зберігати в базі даних всі необхідні дані про стан об'єктів хімічної інфраструктури, включаючи дані про стан обладнання, технологічних процесів, навколишнього середовища та т.д.

Postgres має ряд особливостей, які дозволяють йому ефективно зберігати та обробляти великі обсяги даних. Наприклад, Postgres підтримує такі технології, як:

– логічні розділи;

Це дозволяє розділити базу даних на логічні частини, які можуть бути масштабовані незалежно.

– сторонні таблиці;

Це дозволяє зберігати дані в зовнішніх файлах або базах даних, що може підвищити продуктивність.

– індексування.

Postgres підтримує широкий спектр типів індексів, які можуть бути використані для підвищення продуктивності запитів.

Postgres має широкий спектр функцій безпеки, включаючи:

– аутентифікацію та авторизацію на основі ролей. Це дозволяє контролювати доступ до даних у базі даних;

– шифрування даних. Це дозволяє захистити дані від несанкціонованого доступу;

– механізми захисту від атак. Це дозволяє захистити базу даних від різних видів атак, таких як SQL-ін'єкції та DoS-атаки.

Postgres є однією з найнадійніших СУБД на ринку. Вона має тривалу історію безперебійної роботи, а також підтримується великою спільнотою розробників. Postgres також має широкий спектр функцій безпеки, які дозволяють захистити дані від несанкціонованого доступу.

Postgres може бути масштабовано як горизонтально, так і вертикально. Горизонтальне масштабування дозволяє збільшити продуктивність системи, додаючи додаткові сервери. Вертикальне масштабування дозволяє збільшити продуктивність системи, збільшуючи продуктивність одного сервера.

Postgres є хорошою альтернативою для будь-якої задачі, для якої потрібна СУБД. Однак, для системного аналізу безпеки хімічної інфраструктури Postgres має ряд переваг, які роблять її найкращим вибором [34].:

– здатність зберігати та обробляти великі обсяги даних. Postgres підтримує зберігання даних у масштабі терабайтів і петабайтів, що дозволяє зберігати в базі даних всі необхідні дані про стан об'єктів хімічної інфраструктури;

– надійність та безпека. Postgres має широкий спектр функцій безпеки, включаючи аутентифікацію та авторизацію на основі ролей, шифрування даних та механізми захисту від атак;

– можливість масштабування. Postgres може бути масштабовано як горизонтально, так і вертикально, що дозволяє адаптувати систему до зростаючих потреб.

Postgres є хорошою альтернативою для будь-якої задачі, для якої потрібна СУБД. Однак, для системного аналізу безпеки хімічної інфраструктури Postgres

має ряд переваг, які роблять її найкращим вибором. Postgres має здатність зберігати та обробляти великі обсяги даних, є надійною та безпечною, а також може бути масштабовано як горизонтально, так і вертикально.

3.2 Обґрунтування вибору мови програмування

В умовах постійного розвитку та вдосконалення технологій, питання безпеки та аналізу систем стають критичними, особливо в контексті інфраструктури, пов'язаної з хімічною галуззю. Хімічна індустрія вимагає високого рівня системного аналізу та безпеки для забезпечення ефективного функціонування та запобігання можливим ризикам.

Вибір мови програмування JavaScript для реалізації консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури базується на ряді вагомих факторів, що враховують технічні, екосистемні та стратегічні аспекти.

JavaScript є стандартною мовою для веброзробки, яка забезпечує універсальність та можливість використання її як на клієнтській, так і на серверній стороні. Це дозволяє створювати уніфікований стек технологій, спрощуючи розробку та підтримку.

JavaScript користується великою та активною спільнотою розробників, що забезпечує швидкий обмін ідеями, розв'язання проблем та постійну підтримку. Широка екосистема бібліотек та фреймворків дозволяє вибирати найбільш відповідні інструменти для розв'язання конкретних задач [36].

Асинхронний характер JavaScript в невеликих та середніх проєктів дозволяє оптимально взаємодіяти з різними компонентами системи, забезпечуючи високу відзивчивість, що є критичним у вимірах системного аналізу безпеки.

Використання сучасних інструментів, таких як TypeScript, вдосконалює якість коду, робить його більш стабільним та легко зрозумілим. TypeScript додає статичну типізацію, що сприяє виявленню та усуненню помилок на етапі розробки.

JavaScript забезпечує зручну інтеграцію з різноманітними технологіями, такими як бази даних, мережеві сервіси та інші. Це важливо у випадку розробки комплексної системи для аналізу та забезпечення безпеки інфраструктури хімічної галузі.

React обраний для фронтенду консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури з врахуванням декількох ключових аспектів, які сприяють ефективності та безпеці розробки.

Компонентна архітектура:

– структурованість та підтримка компонентів. React базується на компонентній архітектурі, що полегшує структурування коду та підтримку його частин. Це особливо важливо в системі, яка має багато функціональних частин;

– легкість розширення. Компоненти React легко розширюються та перевикористовуються, що робить систему більш гнучкою та легко розширюваною.

Віртуальний DOM:

– оптимізація швидкодії. Віртуальний DOM дозволяє React оптимально взаємодіяти з реальним DOM, що покращує продуктивність веб-додатку;

– мінімізація перерендерингу. Механізм віртуального DOM дозволяє зменшити перерендеринг елементів, що важливо для забезпечення ефективності та відзивчивості.

Реактивний Підхід:

– спрощення реакції на зміни. Реактивний підхід React дозволяє ефективно реагувати на зміни в стані додатка, що є важливим у вимірах системного аналізу безпеки;

– динамічні інтерфейси. React сприяє створенню динамічних та інтерактивних інтерфейсів, що може бути важливо для представлення інформації щодо системної безпеки.

Безпека:

– контроль стану та даних. React надає можливість забезпечити контроль над станом компонентів та обміну даними між ними, що є критичним для захисту важливої інформації в системі безпеки.

– менша ймовірність уразливостей. З врахуванням активної спільноти, React швидко реагує на знайдені уразливості, зменшуючи ймовірність зовнішніх атак.

Інструментальна підтримка:

– широкі можливості для тестування: React надає добре підтримані засоби для тестування компонентів, що дозволяє забезпечити високу якість коду та системи в цілому.

React обраний для фронтенду проєкт з урахуванням його здатності до структурування та підтримки компонентів, оптимізації продуктивності через віртуальний DOM, реактивного підходу та можливості забезпечити контроль та безпеку системи.

Вибір Koa.js в ролі фреймворку для реалізації бекенду консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури базується на кількох ключових факторах, які забезпечують легкість розробки, асинхронність та високу забезпеченість [37].

Легкість та модульність:

– мінімізація навантаження. Koa.js відомий своєю легкістю та мінімалістичним дизайном, що забезпечує легше управління навантаженням та обслуговуванням запитів, що важливо для систем, де безпека є ключовим аспектом;

– модульність та розширюваність. Модульна архітектура Koa.js дозволяє легко структурувати код та додає нові функціональності без зайвого складнощів.

Асинхронність та Middleware:

– асинхронний характер. Koa.js базується на асинхронному програмуванні та використовує сучасні можливості JavaScript, такі як async/await. Це сприяє ефективній обробці запитів та підвищує продуктивність системи;

– Middleware. Використання middleware дозволяє ефективно обробляти різноманітні аспекти запитів, включаючи безпекові фільтри, логування та автентифікацію.

ES6+ Синтаксис:

– сучасний синтаксис. Використання синтаксису ES6+ забезпечує читабельність та підтримує сучасні стандарти розробки, що важливо для підтримки та розвитку системи в майбутньому.

Підтримка аутентифікації та авторизації:

– легкість інтеграції. Koa.js дозволяє легко інтегрувати та керувати процесами аутентифікації та авторизації, важливими аспектами для систем безпеки.

Підтримка RESTful та GraphQL API:

– гнучкість взаємодії. Підтримка як RESTful, так і GraphQL API надає гнучкість у виборі способу взаємодії з фронтендом та іншими системами.

Вибір JavaScript як основної мови програмування засновано на його універсальності, великій спільноті та активній екосистемі, які забезпечують ефективну розробку та високий рівень підтримки.

React було обрано для фронтенду з урахуванням його компонентної архітектури, віртуального DOM та реактивного підходу. Це дозволяє створювати динамічні та ефективні інтерфейси, які важливі для відображення інформації про системну безпеку.

Вибір Koa.js для бекенду виправданий його легкістю, модульністю та асинхронністю. Його підтримка middleware та ES6+ синтаксису забезпечують ефективну обробку запитів та легкість розширення.

Отже, обрана комбінація JavaScript, React та Koa.js створює ефективний та безпечний технічний стек для розробки консолідованого інформаційного ресурсу для аналізу безпеки хімічної інфраструктури. Це виправдано з точки зору потужності, гнучкості та високого стандарту безпеки в рамках вимог сучасних стандартів веброзробки.

3.3 Практична реалізація бази даних інформаційного ресурсу

Реалізація бази даних для інформаційного ресурсу є важливим етапом у розробці системи.

Для розробки була вибрана система управління базою даних Postgres. Розробимо базу даних за допомогою бібліотеки Express js та бібліотеки Sequelize.

Для початку реалізуємо підключення потрібних модулів та підключення бази даних.

```
const express = require('express');
const { Sequelize, DataTypes } = require('sequelize');
const sequelize = new Sequelize({
 dialect: 'postgres',
 host: '127.0.0.1',
 username: 'your_username',
 password: 'your_password',
 database: 'your_database',});
```

Наступним кроком оголосимо модель користувачів.

```
const User = sequelize.define('User', {
 id: {
  type: DataTypes.INTEGER,
  primaryKey: true,
  autoIncrement: true,  },
 name: {
  type: DataTypes.STRING, },
 email: {
  type: DataTypes.STRING,
  unique: true,  },
 password: {
  type: DataTypes.STRING,  },});
```

Далі оголосимо модель аналізів.

```
const Analysis = sequelize.define('Analysis', {
 id: {
  type: DataTypes.INTEGER,
  primaryKey: true,
  autoIncrement: true,  },
 analysisName: {
  type: DataTypes.STRING,  },
 analysisResult: {
  type: DataTypes.STRING,  },
 analysisDate: {
  type: DataTypes.DATE,  },});
```

Далі оголосимо модель для зберігання інформацію про інфраструктури

```
const ystem = sequelize.define('EnergySystem', {
 id: {
  type: DataTypes.INTEGER,
  primaryKey: true,
  autoIncrement: true,  },
 systemName: {
  type: DataTypes.STRING,  },
 systemDescription: {
  type: DataTypes.STRING,  },
 systemTechnicalSpecification: {
  type: DataTypes.STRING,  },});
```

Далі оголосимо модель для зберігання інформацію про об’єкти інфраструктури.

```
const EnergyObjects = sequelize.define('EnergyObjects', {
 id: {
  type: DataTypes.INTEGER,
  primaryKey: true,
  autoIncrement: true,  },
 objectName: {
  type: DataTypes.STRING,  },
 objectType: {
  type: DataTypes.STRING,  },
 objectLocation: {
  type: DataTypes.STRING,  },});
```

Наступним кроком реалізуємо сутність з засобами забезпечення захисту об’єкта інфраструктури.

```
const SecurityMeasures = sequelize.define('SecurityMeasures', {
 id: {
   type: DataTypes.INTEGER,
   primaryKey: true,
   autoIncrement: true,  },
 measuresDescription: {
   type: DataTypes.STRING,  },
 measuresImplementationDate: {
   type: DataTypes.DATE,  },
 measuresType: {
   type: DataTypes.STRING,  },});
```

Також реалізуємо зв’язки між всіма сутностями бази даних.

```
SecurityMeasures.belongsTo(EnergyObjects);
EnergyObjects.belongsTo(EnergySystem);
EnergyObjects.hasMany(SecurityMeasures);
EnergySystem.hasMany(Analysis);
EnergySystem.hasMany(EnergyObjects);
Analysis.belongsTo(User);
Analysis.belongsTo(EnergySystem);
```

Далі реалізуємо функції, завдяки яким можна буде діставати інформацію з сутностей бази даних для подальшої роботи з нею.

```
app.get('/analyses', async (req, res) => {
 try {
   const analyses = await Analysis.findAll({
     include: [User, EnergySystem], });
   res.json(analyses);
 } catch (error) {
   console.error(error);
   res.status(500).json({ error: 'Internal Server Error' });  }});
app.get('/users', async (req, res) => {
 try {
   const users = await User.findAll();
   res.json(users);
 } catch (error) {
   console.error(error);
```

```
  res.status(500).json({ error: 'Internal Server Error' });  }});
```

Отже, можна зробити висновок, що практична реалізація бази даних пройшла успішно, що дозволяє перейти до наступних етапів розробки.

3.4 Розробка звітів інформаційного ресурсу

Звіти можуть бути різного типу, такого як операційні, стратегічні або тактичні, залежно від мети вивчення інформації. Використання SQL для створення запитів та звітів має свої переваги. SQL забезпечує простий та ефективний засіб взаємодії з базами даних, дозволяючи виражати операції з даними. Ця мова є стандартом для роботи з реляційними базами даних та забезпечує ефективний доступ до даних. Використання SQL також гарантує стандартизацію процесу та сприяє безпеці, завдяки можливості параметризованого використання SQL-запитів для захисту від потенційних атак.

Реалізуємо запит для звіту для відображення рейтингу транспортних інфраструктур по пройденим аналізам безпеки:

```
SELECT
  infrastruture.  infrastruture,
  EnergySystem.systemDescription,
  COUNT(Analysis.id) AS analysisCount
FROM
  infrastruture
LEFT JOIN
  Analysis ON EnergySystem.id = Analysis.EnergySystemId
GROUP BY
  EnergySystem.id, infrastruture.systemName, infrastruture.systemDescription
ORDER BY
  analysisCount DESC;
```

Реалізуємо даний звіт на сторінці користувача (рис. 3.1).

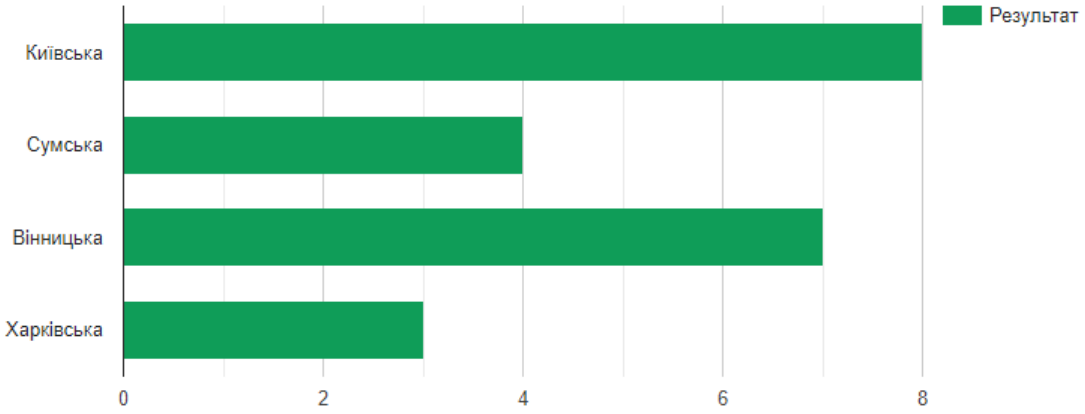


Рисунок 3.1 – Звіт рейтингу інфраструктур за результатами їх аналізів безпеки

Далі реалізуємо запит для звіту рейтинг хімічних інфраструктур за кількістю об’єктів інфраструктури:

```
SELECT
  infrastruture. infrastruture,
  infrastruture.systemDescription,
  COUNT(infrastruture.id) AS ObjectsCount
FROM  infrastrutureLEFT JOIN
  Objects ON infrastruture.id = infrastruture.EnergySystemIdGROUP BY
  infrastruture.id, infrastruture.systemName, infrastruture.systemDescriptionORDER BY
  ObjectsCount DESC;
```

Реалізуємо даний звіт на сторінці користувача (рис. 3.2).

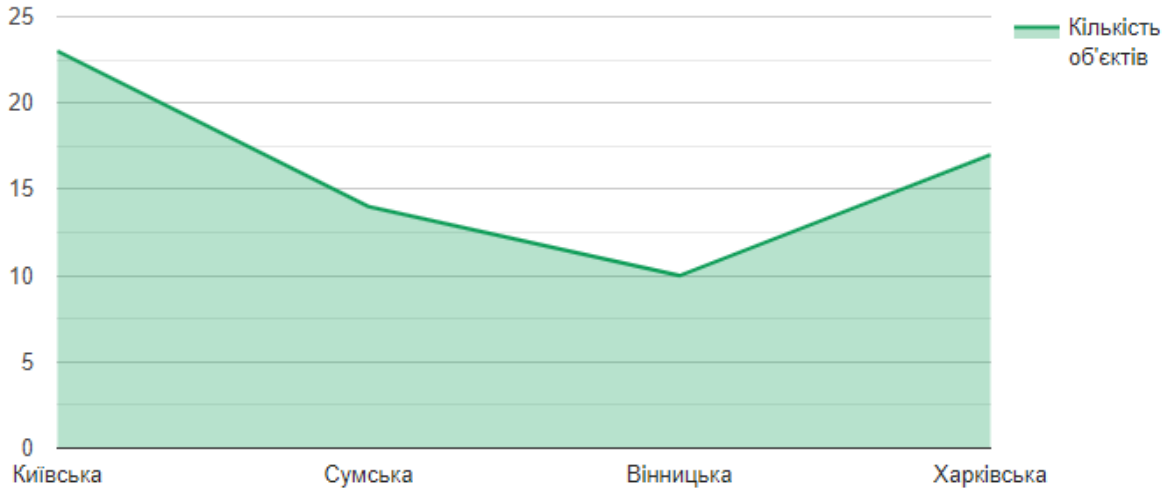


Рисунок 3.2 – Звіт рейтингу хімічних інфраструктур за кількістю об'єктів інфраструктури

Далі реалізуємо запит для звіту рейтинг хімічних об’єктів інфраструктур за кількістю засобів захисту:

```
SELECT
  infrastruture. infrastruture,
  infrastruture.systemDescription,
  COUNT(infrastruture.id) AS ObjectsCount
FROM  infrastrutureLEFT JOIN
  Objects ON infrastruture.id = infrastruture.EnergySystemIdGROUP BY
  infrastruture.id, infrastruture.systemName, infrastruture.systemDescriptionORDER BY
  ObjectsCount DESC;
```

Реалізуємо даний звіт на сторінці користувача (рис. 3.3).

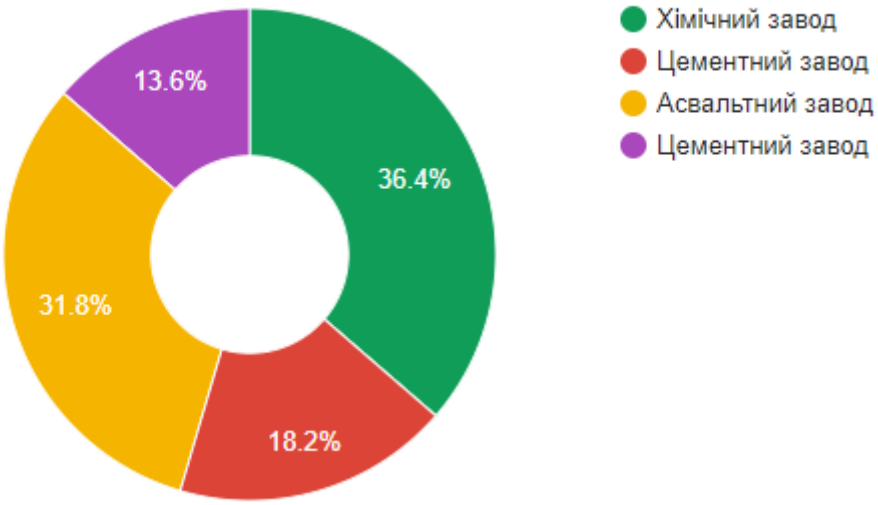


Рисунок 3.3 – Звіт рейтингу хімічних об’єктів інфраструктур за кількістю засобів захисту

Отже, можна зробити висновок, що реалізація звітів та запитів пройшла успішно і користувачі зможуть побачити інформацію в зручному для аналізу вигляді.

3.5 Розробка програмних модулів забезпечення захисту інформаційного ресурсу

Перший кроком в реалізації модулів забезпечення захисту інформаційного ресурсу є ініціалізація проєкт, отже здійснимо ініціалізацію проєкт.

```
const Koa = require('koa');
```

```
const app = new Koa();
app.use(async (ctx) => {
 ctx.body
});
app.listen(3000, () => {
 console.log();
});
```

Наступним кроком реалізуємо функцію варіації вхідних пакетів даних для відфільтровування підозрілих даних.

```
const Joi = require('joi');
const validateInput = (inputData) => {
 const schema = Joi.object({
  name: Joi.string().required(),
 });
 const { error, value } = schema.validate(inputData);
 if (error) {
  throw new Error(`Валідація не пройшла: ${error.message}`); }
 return value;
};try {
 const inputData = {
  name: 'John Doe',  };
 const validatedData = validateInput(inputData);
 console.log('Дані пройшли валідацію:', validatedData);
} catch (error) {
 console.error('Помилка валідації:', error.message);
}
```

Наступним кроком реалізуємо функцію валідування запитів для захисту від SQL-ін’єкцій.

```
const sanitizeInput = (input) => {
 return String(input).replace(/[\0\b\t\n\r\x1a"'\\]/g, function (char) {
  switch (char) {
   case '\0':
    return '\\0';
   case '\b':
    return '\\b';
   case '\t':
    return '\\t';
   case '\n':
    return '\\n';
   case '\r':
    return '\\r';
   case '\x1a':
    return '\\Z';
   case '"':
   case "'":
   case '\\':
    return '\\' + char;
   default:
    return char;   } });};
const sqlInjectionProtection = async (ctx, next) => {
 ctx.sanitize = (input) => sanitizeInput(input);
 ctx.safeQuery = (query, params) => {
  const sanitizedParams = params.map(param => sanitizeInput(param));
  return ctx.db.query(query, sanitizedParams);  };  await next();};
app.use(sqlInjectionProtection);
```

В цьому захисті middleware sqlInjectionProtection додає дві функції до контексту Koa: ctx.sanitize та ctx.safeQuery. Функція ctx.sanitize екранує спеціальні символи у рядках, щоб убезпечити від SQL-ін'єкцій. Функція ctx.safeQuery використовує цю екранізацію для параметрів запиту перед їх передачею в базу даних.

При використанні цих функцій, важливо не використовувати ручне конструювання SQL-рядків із введеними користувачем даними та завжди використовувати параметризовані запити або ORM. Цей захист спрямований на екранування та очищення даних перед їх використанням в SQL-запитах.

Наступним кроком реалізуємо фаєрвол, який буде відфільтровувати шкідливий трафік.

```
 return allowedIPs.includes(ip);};
const checkURLAccess = (url) => {
 const allowedURLs = ['/public', '/open'];
 return allowedURLs.some(allowedURL => url.startsWith(allowedURL));};
const firewallMiddleware = async (ctx, next) => {
 const clientIP = ctx.request.ip;
 if (!checkIPAccess(clientIP)) {
  ctx.status = 403; // Заборонити доступ
  ctx.body = 'Доступ заборонено';
  return;
 } const requestURL = ctx.request.url;
 if (!checkURLAccess(requestURL)) {
  ctx.status = 403; // Заборонити доступ
  ctx.body = 'Доступ заборонено';
  return;
 } await next();};app.use(firewallMiddleware);
```

Наступним кроком реалізуємо функцію запам’ятовування джерела шкідливого трафіку для подальшого його фільтрації.

```
const suspiciousIPs = new Set();
const addSuspiciousIP = (ip) => {
 suspiciousIPs.add(ip);
 console.log(`Додано IP в список підозрілих: ${ip}`);};
const firewallMiddleware = async (ctx, next) => {
 const isSuspiciousTraffic =
```

```
 if (isSuspiciousTraffic) {    addSuspiciousIP(ctx.request.ip);  }
 await next();};
app.use(firewallMiddleware);
app.use(async (ctx) => {
 ctx.body = 'Ласкаво просимо!';});
```

Отже, можна зробити висновок, що реалізація модулів захисту пройшла успішно, даний захист дозволить захистити інформаційний ресурс від шкідливого трафіку та від SQL-інєєкцій.

3.6 Реалізація програмних засобів системного аналізу безпеки хімічної інфраструктури регіону

Початковим кроком в реалізації системи аналізу безпеки хімічної інфраструктури регіону є реалізація логіки роботи системи. Отже, спочатку реалізуємо логіку додавання інфраструктури.

```
import { Module } from '@nestjs/common';
import { TypeOrmModule } from '@nestjs/typeorm';
import { Infrastructure } from './infrastructure.entity';
import { InfrastructureService } from './infrastructure.service';
import { InfrastructureController } from './infrastructure.controller';
@Module({
 imports: [TypeOrmModule.forFeature([Infrastructure])],
 providers: [InfrastructureService],
 controllers: [InfrastructureController],
})
 @Post()
 async createInfrastructure(@Body() createInfrastructureDto: { name: string; description: string }) {
  const { name, description } = createInfrastructureDto;
  return this.infrastructureService.createInfrastructure(name, description);
 }
export class AppModule {}
```

Далі реалізуємо інтерфейс на сторінці користувача.

```
<div class="container">
  <div class="form-container" id="login-form">
    <h1>Додавання інфраструктури</h1>
    <form>
      <label for="password">Назва</label>
      <input type="password" id="password" name="password" required>
```

```
      <label for="password">Опис</label>
    <input type="password" id="password" name="password" required>
      <label for="password">Специфікації</label>
    <input type="password" id="password" name="password" required>
    <button type="submit">Додати</button>
  </form>
 </div>
```

В результаті чого була отримана наступна сторінка (рис. 3.4).

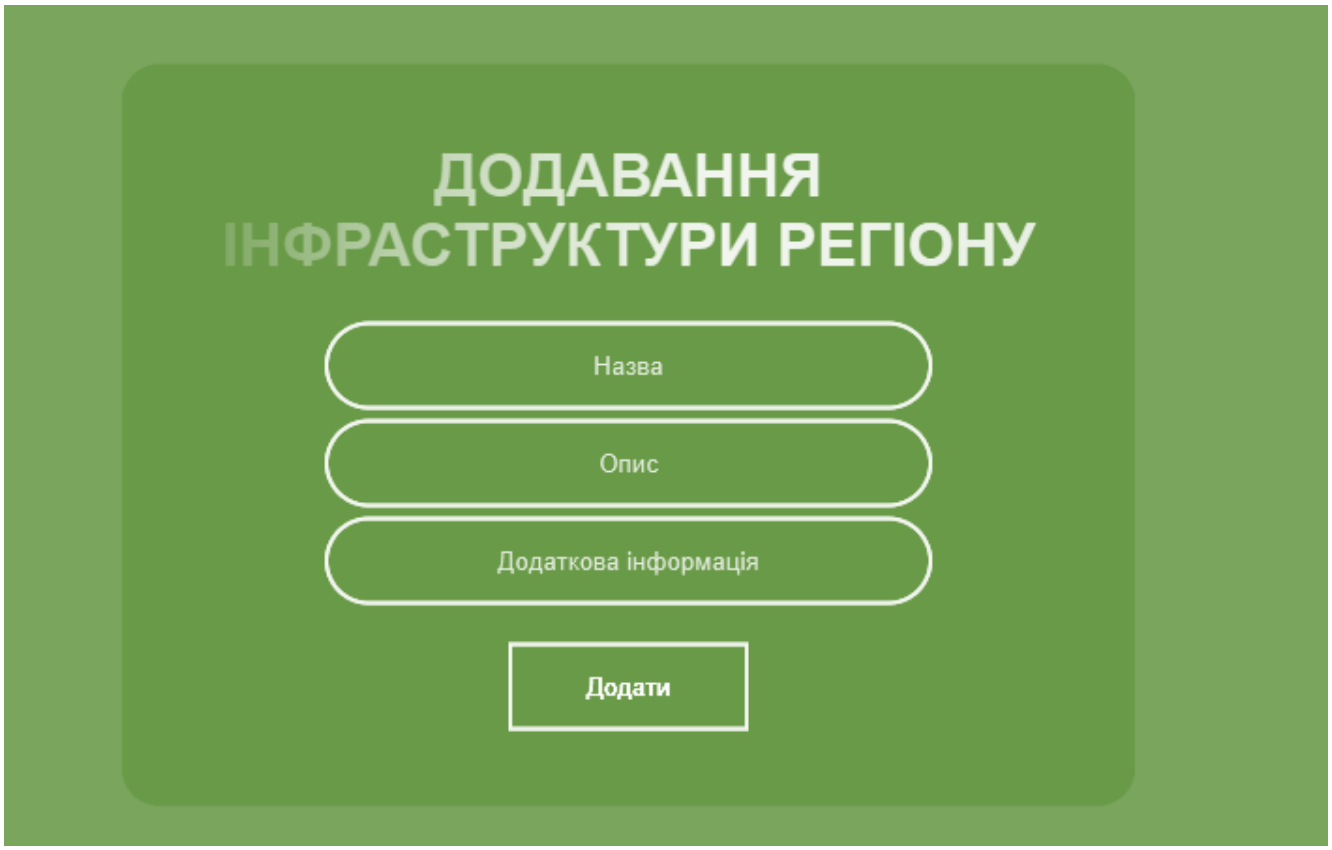


Рисунок 3.4 – Сторінка додавання інфраструктури

Наступним кроком реалізуємо логіку додавання об’єктів інфраструктури.

```
router.post('/add', async (req, res) => {
 try {
  const { objectName, objectType, objectLocation, systemId } = req.body;
  const newObject = await EnergyObjects.create({
   objectName,
   objectType,
   objectLocation,
   systemId,   });
  res.status(201).json({ message: 'Об\'єкт інфраструктури додано успішно', object: newObject });
```

```
 } catch (error) {
   console.error(error);
   res.status(500).json({ message: 'Помилка сервера' });  }});
```

Також реалізуємо відповідну сторінку на стороні користувача.

```
<div class="container">
   <div class="form-container" id="login-form">  <h1>Додавання об'єкта інфраструктури</h1>    <form>
      <label for="password">Назва</label>
      <input type="password" id="password" name="password" required>
        <label for="password">Локація</label>
      <input type="password" id="password" name="password" required>
        <label for="password">Тип</label>
      <input type="password" id="password" name="password" required>
      <button type="submit">Додати</button>     </form>    </div>
   <div class="form-container" id="signup-form" style="display: none;">
```

В результаті була отримана наступна сторінка (рис 3.5).

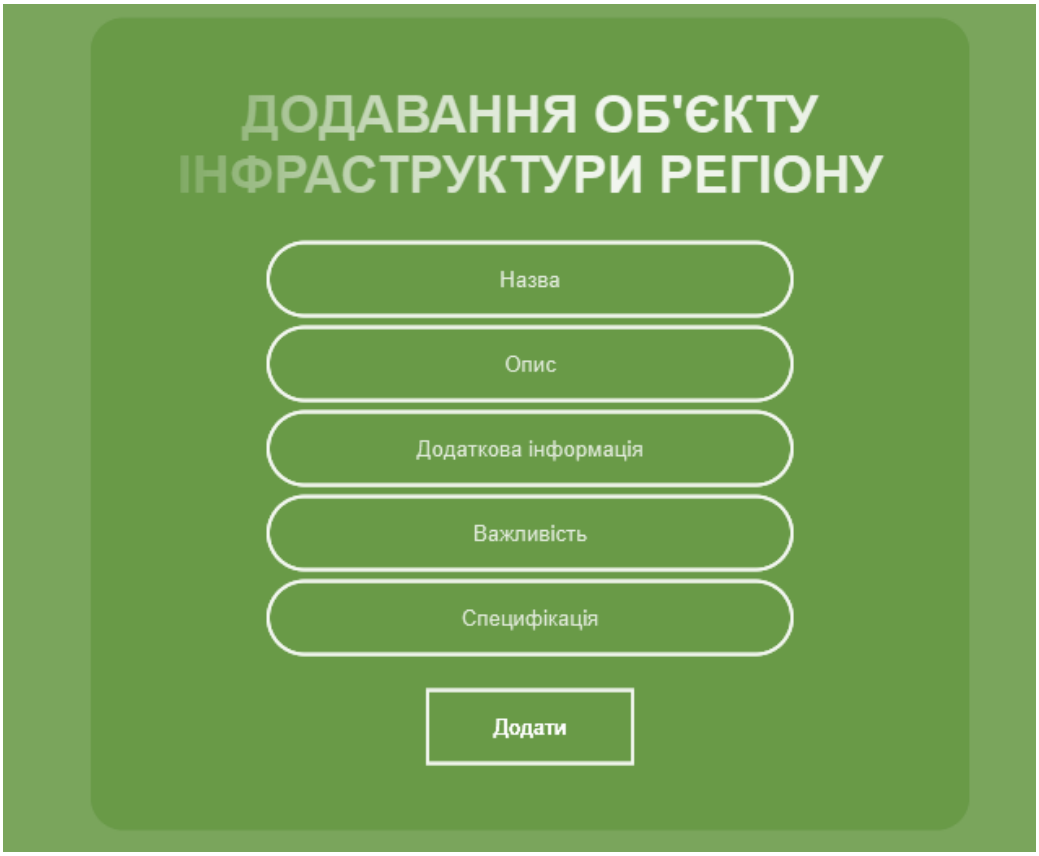


Рисунок 3.5 – Сторінка додавання об’єкта інфраструктури

Далі реалізуємо логіку додавання методів захисту об’єктів інфраструктури.

```
router.post('/add', async (req, res) => {
 try {   const { measuresDescription, implementationDate, measuresType, objectId } = req.body;
```

```
    const newMeasure = await SecurityMeasures.create({
      measuresDescription,
      implementationDate,
      measuresType,
      objectId,    });
    res.status(201).json({ message: 'Засіб захисту додано успішно', measure: newMeasure });
  } catch (error) {
    console.error(error);
    res.status(500).json({ message: 'Помилка сервера' });  }});
```

Також реалізуємо відповідну сторінку на стороні користувача.

```
<div class="container">
    <div class="form-container" id="login-form">
      <h1>Додавання засобів захисту</h1>
      <form>
        <label for="password">Назва</label>
        <input type="password" id="password" name="password" required>
          <label for="password">Дата</label>
        <input type="password" id="password" name="password" required>
          <label for="password">Тип</label>
        <input type="password" id="password" name="password" required>
           <label for="password">Опис</label>
        <input type="password" id="password" name="password" required>
        <button type="submit">Додати</button>
      </form>
    </div>
```

В результаті була отримана наступна сторінка (рис 3.6).

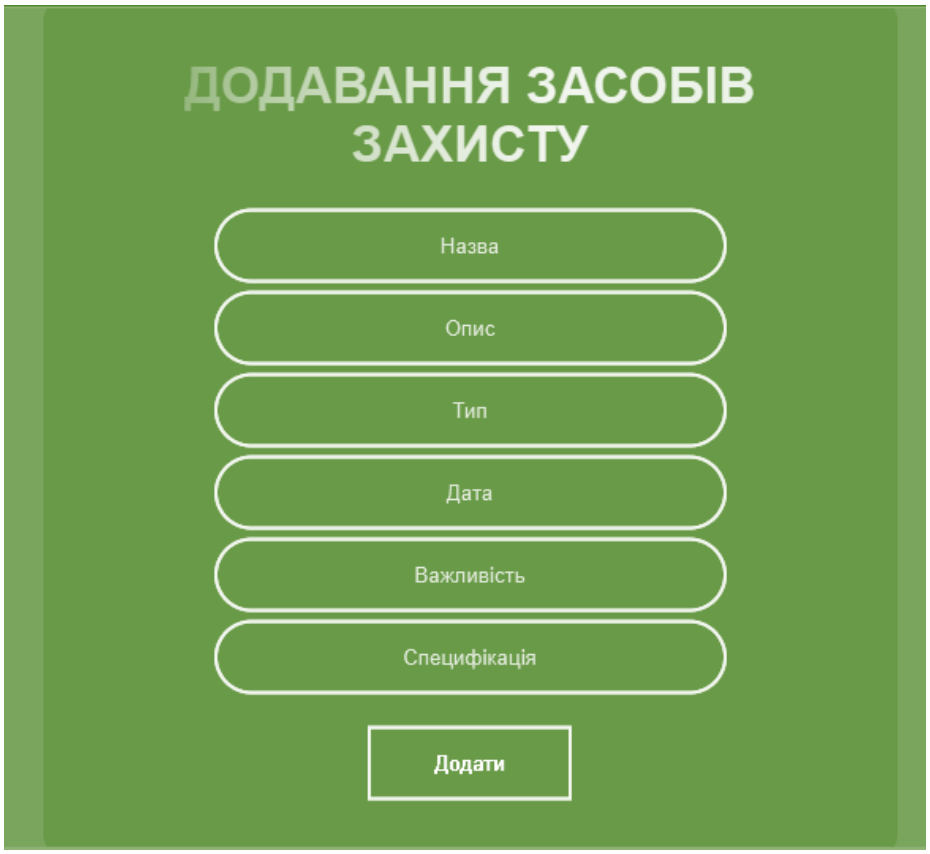


Рисунок 3.6 – Сторінка додавання систем захисту об’єкта

Наступним кроком реалізуємо логіку оцінювання безпеки хімічної інфраструктури регіону. Спочатку реалізуємо логіку проходження питань (рис. 3.4).

```
 const app = express();
app.use(bodyParser.json());
app.get('/questions', (req, res) => {
  res.json(questions);});
app.post('/submit', (req, res) => {
  const userAnswers = req.body.answers;
  let score = 0;
  questions.forEach((question, index) => {
    if (userAnswers[index] === question.correctOption) {
      score++;  } });
  res.json({ score, totalQuestions: questions.length });});
app.listen(PORT, () => {
```

Також реалізуймо відповідну логіку та сторінку на стороні користувача.

```
<script>
```

```
async function fetchQuestions() {
  const response = await fetch('/questions');
  const questions = await response.json();
  return questions   }
async function displayQuestions() {
  const questions = await fetchQuestions();
  const container = document.getElementById('questions-container');
  questions.forEach((question, index) => {
    const questionElement = document.createElement('div');
    questionElement.innerHTML = `
      <p>${index + 1}. ${question.text}</p>    <ul>
        ${question.options.map((option,    optionIndex)    =>    `<li><input    type="radio"    name="q${index}"
value="${optionIndex}">${option}</li>`).join('')}
      </ul>     container.appendChild(questionElement);     });
async function submitAnswers() {     const userAnswers = [];
  const answerInputs = document.querySelectorAll('input[type="radio"]:checked');
  answerInputs.forEach(input => {
    userAnswers.push(parseInt(input.value)); });
  const response = await fetch('/submit', {
    method: 'POST',       headers: {        'Content-Type': 'application/json',   },
   body: JSON.stringify({ answers: userAnswers }),   });
  const result = await response.json();
  alert(`Ваш результат: ${result.score}/${result.totalQuestions}`); }
```

В результаті реалізації була отримана наступна сторінка (рис. 3.7).

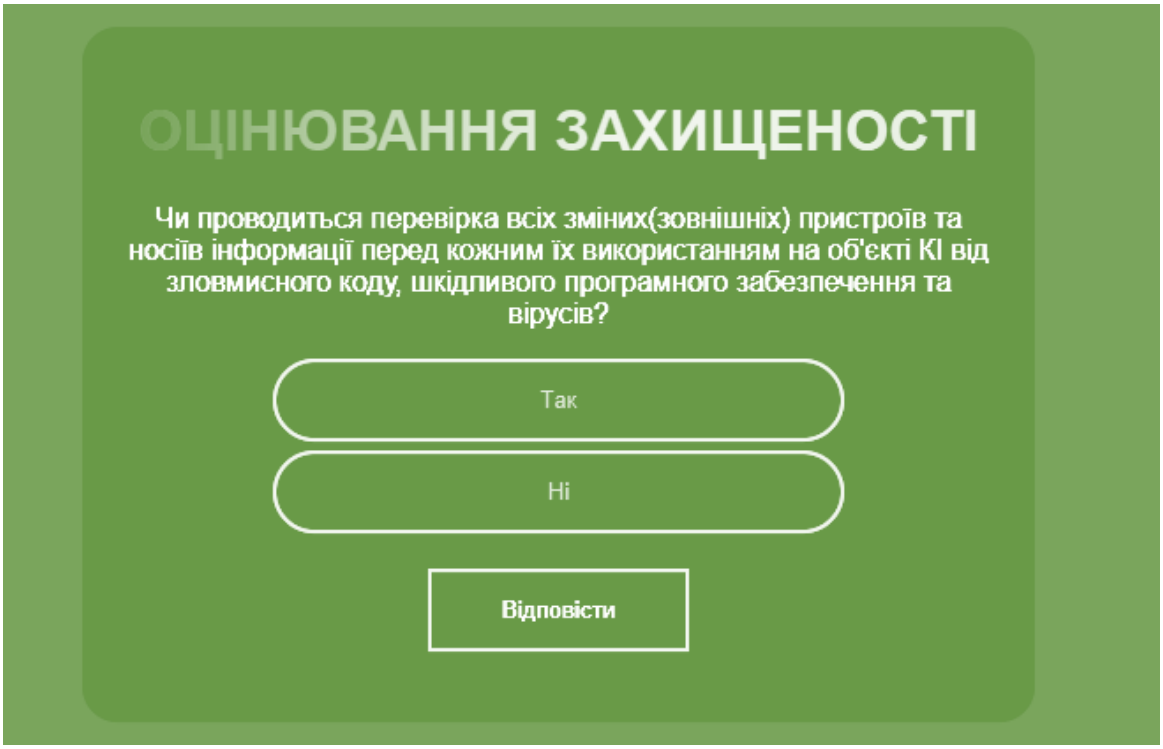


Рисунок 3.7 – Сторінка проходження оцінювання безпеки

Наступним кроком реалізуємо логіку оцінювання та запису результатів в базу даних.

```
app.post('/save-result', async (req, res) => {
 try {
  const { userId, score, totalQuestions } = req.body;
  const result = await TestResult.create({
   userId,
   score,
   totalQuestions,   });
  res.status(201).json({ message: 'Результат тестування збережено успішно', result });
 } catch (error) {
  console.error(error);
  res.status(500).json({ message: 'Помилка сервера' });  }});
```

Також реалізуємо сторінку виведення результатів.

```
<body>
 <h1>Результати тестування</h1>
 <% if (results.length === 0) { %>  <p>Немає результатів тестування.</p>
 <% } else { %>   <table>    <thead>
     <tr>     <th>ID</th>
      <th>User ID</th>
```

```
        <th>Score</th>
        <th>Total Questions</th> </tr>     </thead>
    <tbody>  <% results.forEach(result => { %>  <tr>
        <td><%= result.id %></td>
        <td><%= result.userId %></td>
        <td><%= result.score %></td>
        <td><%= result.totalQuestions %></td>  </
<% }); %>  </tbody>   </table>  <% } %>
```

В результаті реалізації була отримана наступна сторінка (рис. 3.8).

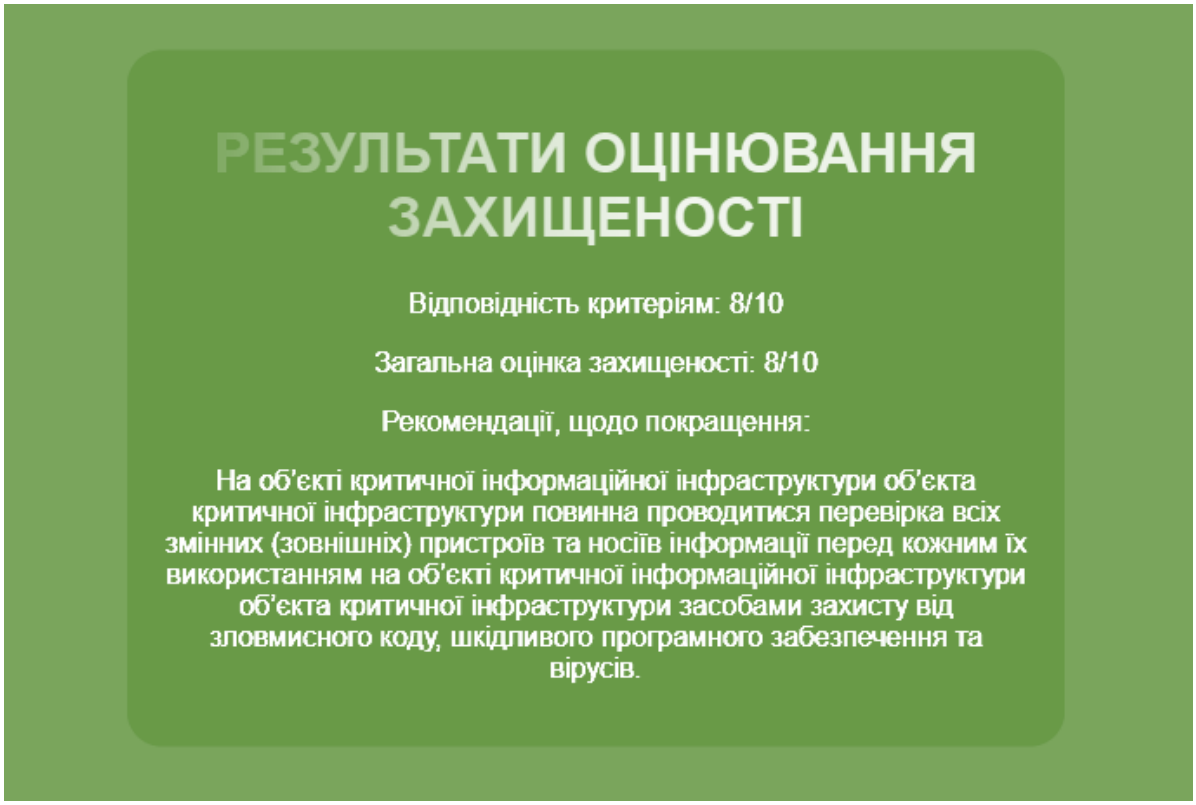


Рисунок 3.8 – Сторінка результатів

Отже, можна зробити висновок, що реалізація пройшла успішно, в результатів були створено логіка роботи системи оцінювання безпеки, також були створені сторінки для зручного користування системою.

3.7 Висновки до розділу

У ході даного дослідження та розробки були визначені, обґрунтовані та реалізовані ключові аспекти створення та захисту інформаційного ресурсу.

Обрана система управління базами даних PostgreSQL виявилася ефективною з точки зору надійності та можливостей масштабування.

Вибір мови програмування JavaScript для реалізації серверної частини обґрунтовується широким використанням у галузі веброзробки та можливістю використання на обох сторонах вебдодатка.

Розроблена база даних інформаційного ресурсу використовує реляційні таблиці та зовнішні ключі для ефективного управління та зберігання великих обсягів даних.

Забезпечення інформаційного ресурсу вбудовано через використання моделі ролей, а також механізмів автентифікації та авторизації, що гарантує конфіденційність та цілісність інформації.

Обрані технології та підходи були обґрунтовані їхньою ефективністю, актуальністю та відповідністю завданням інформаційного ресурсу.

У висновку можна стверджувати, що розроблені рішення сприятливо впливають на якість та безпеку управління інформаційними ресурсами, забезпечуючи оптимальні умови для їхньої роботи та розвитку в сучасному інформаційному середовищі.

4 ЕКОНОМІЧНА ЧАСТИНА

У даному розділі ми проведемо аналіз економічного потенціалу розробки, що включатиме оцінку комерційних можливостей, прогнозування витрат на виконання наукової роботи та впровадження її результатів, а також розрахунок прогнозованих комерційних вигід від реалізації розробленого продукту та визначення ефективності вкладених інвестицій і часу їх повернення. На підставі отриманих результатів аналізу буде зроблено висновок щодо економічної доцільності розробки захищеного консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури регіону.

4.1 Оцінювання комерційного потенціалу розробки програмного забезпечення

Метою здійснення технологічного аудиту є оцінка комерційного потенціалу, що випливає з результатів науково-технічної діяльності.

Результатом виконання магістерської кваліфікаційної роботи є створення захищеного консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури регіону, який практично реалізований у вигляді веб-сервісу.

Для проведення технологічного аудиту були залучені три незалежні експерти.

У межах даної роботи такими експертами є викладачі кафедри МБІС:

– Карпінець В. В. (к.т.н., доцент каф. МБІС ВНТУ);
– Яремчук Ю. Є. (д.т.н., проф. МБІС ВНТУ);
– Грицак А. В. (доц., викл. каф. МБІС ВНТУ).

Оцінювання комерційного потенціалу здійснимо за критеріями, що наведені у 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

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
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

Необхідно вести результати оцінки науково-технічного рівня та комерційного потенціалу науково-технічної розробки у форму таблиці.

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

Критерії	Прізвище, ініціали, посада експерта		
	1 – Карпінець В. В.	2 – Яремчук Ю. Є.	3 – Грицак А. В.
1	4	4	4

Продовження таблиці 4.2

Ринкові переваги (недоліки):			
2	4	3	4
3	3	4	3
4	4	4	4
5	4	3	4
Ринкові перспективи			
6	3	4	4
7	4	3	4
Практична здійсненність			
8	3	4	3
9	4	3	4
10	3	4	4
11	4	4	4
12	4	3	3
Сума балів	$СБ_1 = 44$	$СБ_1 = 43$	$СБ_1 = 45$
Середньоарифметична сума балів $СБ_с$	$СБ = 44$		

На основі даних, представлених у таблиці 4.2, можна здійснити висновок щодо рівня комерційного потенціалу розробки. Порівняємо отримані результати з рівнями комерційного потенціалу, які відображені 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

На основі проведених досліджень встановлено, що рівень комерційного потенціалу розробки, яка присвячена темі "Захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону", становить 44 бали. Згідно з інформацією у таблиці 4.3, це свідчить про високу комерційну важливість проведення цих досліджень.

4.2 Прогнозування витрат на виконання наукової роботи та впровадження її результатів

Процес прогнозування витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технічної роботи складається із трьох ключових етапів, кожен із яких детально розглядає різні аспекти витрат, що впливають на всі етапи виконання проекту.

На першому етапі проводиться аналіз витрат, які прямо пов'язані із зусиллями та ресурсами, витраченими виконавцями цього розділу роботи. Це включає оплату праці, витрати на навчання та інші витрати, що безпосередньо пов'язані із виконанням цієї конкретної роботи.

Другий етап передбачає розрахунок загальних витрат на виконання всієї роботи, охоплюючи витрати на матеріали, обладнання, послуги та інші загальні витрати, пов'язані із всім проектом.

Третій етап включає в себе прогнозування загальних витрат на виконання впровадження результатів даної роботи. Це охоплює витрати на впровадження розробок, рекламу, підготовку персоналу та інші витрати, пов'язані із реалізацією отриманих результатів.

Важливо відзначити, що на даному етапі використовується конкретна структура витрат, враховуючи, що для розробки інформаційної технології застосовується лише один розробник програмного забезпечення та керівник проекту.

Основна заробітна плата $З_о$:

$$З_о = \frac{М}{Т_р} \times t, \text{грн}$$

(4.1)

Де М – місячний посадовий оклад;

$Т_р$ – число робочих днів в місяць; приблизно $Т_р = 22$ днів;

t – число робочих днів роботи – 58 днів.

Таким чином:

$$З_о = \frac{25\,000}{22} \times 58 = 65\,909(\text{грн.})$$

Таблиця 4.4 – Витрати по заробітній платі

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату
Керівник проекту	25 000	1136,4	58	65 909
Розробник програмного забезпечення	21 000	954,5	50	47 727,3
Тестувальник програмного забезпечення	17 000	772,73	10	7 727,3
Всього				121 363,6

Додаткова заробітна плата $З_д$ працівників розраховується як 11% від основної заробітної плати:

$$З = 0{,}11 \times 121\,363{,}6 = 13\,350\ (\text{грн.})$$ – для розробника

Нарахування на заробітну плату $Н_{зп}$ розробника становить:

$$Н_{зп} = (З_о + З_д) \times \frac{\beta}{100}$$

(4.2)

$З_о$ – основна заробітна плата розробника;

$З_д$ – додаткова заробітна плата розробника;

β – ставка єдиного внеску на загальнообов'язкове державне соціальне страхування – 22%

$$\text{Н}_{\text{зп}} = (121\,363{,}6 + 13\,350) \times 0{,}22 = 29\,637\ (\text{грн.})$$

Стаття витрат «Програмне забезпечення» належать витрати на придбання необхідного програмного забезпечення та витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{прг} = \sum_{i=1}^{k} Ц_{iпрг} \cdot C_{прг.i} \cdot K_i \ , \qquad (4.3)$$

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, ($K_i = 1{,}10...1{,}12$);

k – кількість найменувань програмних засобів.

$\text{Впрг} = 6\,042 \cdot 1 \cdot 1{,}1 = 6\,646{,}2$ *грн.*

Отримані результати зведемо до таблиці:

Таблиця 4.5 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
ОС Windows 11	1	5 250	5 775
IDE WebStorm	1	6 042	6 646,2
Всього			12 421,2

Амортизація обладнання, комп'ютерів та приміщень, які використовувались під час виконання даного етапу роботи розраховуємо за формулою:

$$\text{А} = \frac{\text{Ц} \times \text{Т}}{12 \times \text{Т}_{\text{в}}} \qquad (4.4)$$

Ц – загальна балансова вартість обладнання, приміщення тощо, грн.;

Т – фактична тривалість використання, міс;

$Т_о$ – термін використання обладнання, приміщення тощо, роки.

Розробка програмного забезпечення ведеться приблизно 2 місяці.

Для офісного приміщення $А = \frac{47\,000 \times 2}{12 \times 20} = 391{,}6$ грн.;

Для ноутбука $А = \frac{22\,999 \times 2}{12 \times 4} = 958{,}3$ грн.;

Розрахунки зведено до таблиці 4.5:

Таблиця 4.6 – Амортизаційні відрахування

Найменування	Балансова вартість (грн.)	Термін використання (років)	Фактична тривалість використання, (міс.)	Величина амортизаційних відрахувань, (грн.)
Офісне приміщення	47000	20	2	391,6
Ноутбук Acer Aspire 5 A515-48M-R1YX	22 999	4	2	958,3
Ноутбук realme Book Prime I5 512GB	25 999	4	2	1 083,3
ОС Windows 11	5 250	2	2	437,5
IDE WebStorm	6 042	1	2	1007
			Всього	3 877,7

Витрати на матеріали, що були використані під час виконання даного етапу роботи, розраховуються за формулою:

$$К = \sum_{1}^{п} Н_i \times Ц_i \times К_i - \sum_{j=1}^{n} В_j \times Ц_{в\ j}\ (грн.)$$

(4.5)

$Н_j$ – 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, (Кj = 1,1 … 1,15);

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

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

Таблиця 4.7 – Витрати на матеріали

Найменування матеріалів	Ціна за од, грн	Норма витрат, од	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Комп'ютерна мишка	480 грн.	2	0	0	1056 грн..
Килимок для миші	675 грн.	1	0	0	742,5 грн.
Папір для записів	90 грн	1	0	0	99 грн
Офісний органайзер	150 грн	2	0	0	330 грн
Настільний канцелярський набір	265 грн.	2	0	0	583 грн
				Всього	2 810,5 грн.

Витрати на силову електроенергію $В_е$ розраховуються за формулою:

$$В_е = \sum_{i=1}^{п} \frac{W_{yt} \times t_i \times Ц_в \times К_{впі}}{\eta_i}\ (грн.) \qquad (4.6)$$

$Ц_в$ – вартість 1 кВт – год. (на сьогодні для підприємців вартість 7,5 грн./кВт-год.);

W_{yt} – установлена потужність обладнання;

t_i – фактична кількість годин роботи обладнання;

$K_{вni}$ – коефіцієнт використання потужності, $K_{вni} < 1$;

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

Таблиця 4.8 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Ноутбук Acer Aspire 5 A515-48M-R1YX	0,6 кВт	406 год.	1 789,3 грн.
Ноутбук realme Book Prime I5 512GB	0,6 кВт	350 год.	1 542,5
Робоче місце розробника	0.04 кВт	406 год.	119,3 грн.
		Всього	3 451,1 грн.

Інші витрати $В_{ін}$ охоплюють:

– витрати на управління організацією;

– оплату службових відряджень;

– витрати на утримання, ремонт та експлуатацію, основних засобів;

– витрати на опалення, водопостачання, охорону праці тощо.

Інші витрати $В_{ін}$ можна прийняти як 100% від суми основної заробітної плати працівників:

$$В_{ін} = 121\,363{,}6 \times 1 = 121\,363{,}6\ (\text{грн.})$$

Послуги Інтернету – 375 грн.

Сума всіх попередніх статей витрат дає витрати на виконання даної частини роботи – В.

$$В = 121\,363{,}6 + 13\,350 + 29\,637 + 12\,421{,}2 + 3\,877{,}7 + 2\,810{,}5 + 3\,451{,}1 + 121\,363{,}6 + 375 = 308\,649{,}7\ (\text{грн.})$$

Проведемо прогнозування загальних витрат ЗВ на виконання та впровадження виконаної наукової роботи. Прогнозування здійснюється за формулою:

$$ЗВ = \frac{В_{заг}}{\beta}, \text{грн.} \tag{4.7}$$

β – коефіцієнт, який характеризує етап виконання даної роботи.

Так, якщо розробка знаходиться:

- на стадії науково-дослідних робіт, то $\beta \approx 0{,}1$;
- на стадії технічного проєктування, то $\beta \approx 0{,}2$;
- на стадії розробки конструкторської документації, то $\beta \approx 0{,}3$;
- на стадії розробки технології, то $\beta \approx 0{,}4$;
- на стадії розробки дослідного зразка, то $\beta \approx 0{,}5$;
- на стадії розробки промислового зразка, то $\beta \approx 0{,}7$;
- на стадії впровадження, то $\beta \approx 0{,}9$.

$В_{заг}$ – загальна вартість всієї наукової роботи.

$$В = 308\,649{,}7\ (\text{грн.})$$

$$ЗВ = \frac{308\,649{,}7}{0.7} = 440\,928{,}14\ (\text{грн.})$$

Отже, передбачена сума загальних витрат (ЗВ) на виконання та впровадження результатів виконаної роботи становить 440 928,14 (грн).

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

У даному розділі представлено кількісний прогноз та оцінку очікуваної користі та потенційного прибутку від впровадження результатів наукової роботи у майбутньому.

У сучасних умовах ринкової конкуренції важливим показником позитивного впливу, який підприємство отримує від впровадження розробок, є зростання чистого прибутку. Визначення цього зростання може бути оцінене в теперішній вартості грошей.

Збільшення чистого прибутку внаслідок впровадження розробки вплине на надходження додаткових коштів для підприємства, сприяючи поліпшенню фінансових показників його діяльності. Тривалість реалізації наукової роботи та впровадження результатів оцінюється приблизно у 7 місяців. Позитивні результати від впровадження розробки очікуються навіть у перший місяць після впровадження.

Цей період впровадження не лише дозволить отримати перші позитивні вигоди, але й створить передумови для стійкого зростання прибутковості підприємства в подальшому. Реалізація проекту буде відзначена швидким ефектом і позитивним внеском у фінансовий успіх підприємства.

Далі проведемо докладне прогнозування позитивних результатів та їх кількісне оцінювання по роках. Збільшення чистого прибутку підприємства (ΔП) для кожного із років, протягом яких передбачається отримання позитивних результатів від впровадження розробки, розраховується за певною формулою:

$$\Delta П_i = \sum_{1}^{n} (\pm \Delta Ц_о \times N + Ц_о \times \Delta N)_i \times \lambda \times \rho \times (1 - \frac{\vartheta}{100}),$$

(4.8)

$\pm\Delta Ц_о$ – зміна вартості програмного продукту від впровадження результатів науково-технічної розробки, приймемо 11000,00 грн;

N – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки, приймемо 200 користувачів;

$Ц_б$ – вартість програмного продукту у році до впровадження результатів розробки, приймемо 550 000,00 грн;

ΔN – збільшення кількості споживачів продукту, у періоди часу, що аналізуються, від покращення його певних характеристик;

Показник	1-й рік	2-й рік	3-й рік
Збільшення кількості споживачів, проектних груп	20	30	50

λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2023 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0{,}8333$;

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

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році $\vartheta = 18\%$;

Збільшення чистого прибутку $\Delta П_1$ протягом першого року складе:

$$\Delta П_1 = (11000 \times 200 + 561\,000 \times 20) \times 0{,}83 \times 0{,}3 \times \left(1 - \frac{0{,}18}{100}\right) = 3\,335\,565{,}2\ (\text{грн.})$$

Обчислимо збільшення чистого прибутку $\Delta П_2$ протягом другого року:

$$\Delta П_2 = \left(11000 \times 200 + 561\,000 \times (20 + 30)\right) \times 0{,}83 \times 0{,}3 \times \left(1 - \frac{0{,}18}{100}\right) = 7\,518\,692\ (\text{грн.})$$

Збільшення чистого прибутку $\Delta П_3$ протягом третього року становитиме:

$$\Delta П_З = \left(11000 \times 200 + 561\,000 \times (20 + 30 + 50)\right) \times 0{,}83 \times 0{,}3 \times \left(1 - \frac{0{,}18}{100}\right)$$
$$= 14\,490\,569{,}9\ (\text{грн.})$$

Отже, відповідно до обчислень, комерційна вигода від впровадження розробки, як і передбачалося, буде суттєвою і виявиться в зростанні чистого прибутку підприємства.

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Ключовими показниками, які визначають призначеність фінансування певним інвестором наукової розробки, є абсолютна та відносна ефективність вкладених коштів і термін їх окупності.

На першому етапі проводиться обчислення теперішньої вартості інвестицій (PV), вкладених у наукову розробку.

Величина початкових інвестицій PV, які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$\text{PV} = k_{інв} \times ЗВ \tag{4.9}$$

$k_{інв}$ – 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 440 928,14 грн.

$$\text{PV} = 3 \times 440\,928{,}14 = 1\,322\,784{,}4$$

На другому етапі виконується розрахунок очікуваного зростання прибутку ($\Delta П_і$), яке підприємство (організація) отримає від впровадження результатів

наукової розробки. Цей розрахунок проводиться для кожного року, починаючи з першого року впровадження.

Таке збільшення прибутку також було розраховане нами раніше та становить:

$$\Delta П_1 = 3\ 335\ 565{,}2\ \ (\text{грн.})$$
$$\Delta П_2 = 7\ 518\ 692\ (\text{грн.})$$
$$\Delta П_3 = 14\ 490\ 569{,}9\ (\text{грн.})$$

Третій етап передбачає створення вісі часу, на якій будуть відображені всі фінансові потоки, включаючи інвестиції та прибутки, які виникають під час виконання науково-дослідної роботи та впровадження її результатів.

Рисунок 4.1 характеризує рух платежів (інвестицій та додаткових прибутків).

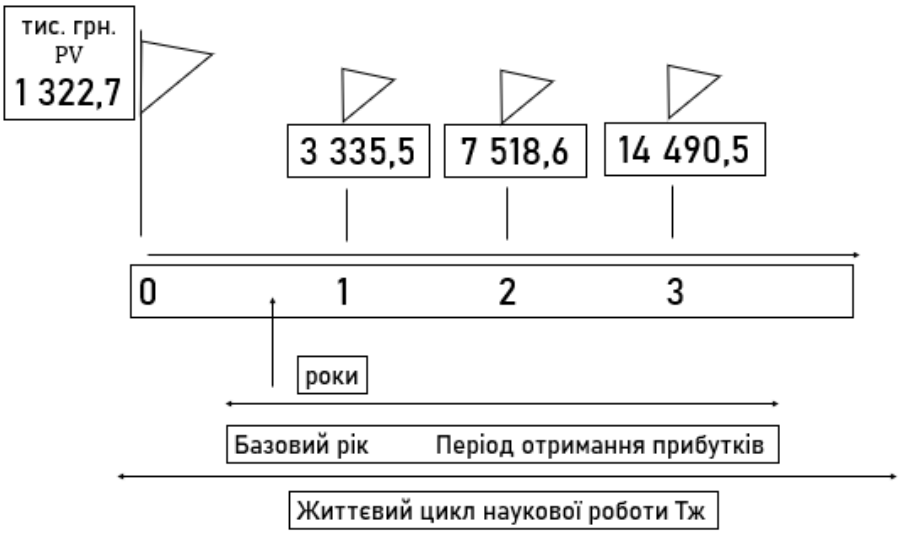


Рисунок 4.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

На четвертому етапі здійснюємо розрахунок абсолютної ефективності вкладених інвестицій ($Е_{абс}$) за допомогою визначеної формули.

$$Е_{абс} = (ПП - PV), (\text{грн.})$$

(4.10)

ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн.;

Приведена вартість всіх чистих прибутків ПП розраховується за формулою:

$$ПП = \sum_{1}^{T} \frac{\Delta П_1}{(1+\tau)^t}, (грн) \quad (4.11)$$

$\Delta П_1$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн,;

Т – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні – 0,15;

t – період часу (в роках) від моменту отримання чистого прибутку до точки «0»;

$$ПП = \frac{3\ 335\ 565{,}2}{(1+0{,}15)^1} + \frac{7\ 518\ 692}{(1+0{,}15)^2} + \frac{14\ 490\ 569{,}9}{(1+0{,}15)^3} = 18\ 113\ 500\ (грн.)$$

$$Е_{абс} = 18\ 113\ 500 - 1\ 322\ 784{,}4 = 16\ 790\ 715{,}6\ (грн.)$$

Оскільки $Е_{абс}$ перевищує 0, встановлено, що виконання наукових досліджень для розробки програмного продукту та його подальше впровадження призведе до отримання прибутку. Це свідчить про доцільність проведення досліджень. Проте цей факт не гарантує зацікавленості інвестора у фінансуванні даної програми.

На п'ятому етапі розраховуємо відносну (щорічну) ефективність вкладених інвестицій у наукову розробку ($Е_в$) за визначеною формулою.

$$Е_в = \sqrt[Т_ж]{1 + \frac{Е_{абс}}{PV}} - 1$$

(4.12)

$Е_{абс}$ – абсолютна ефективність вкладених інвестицій, грн.;

PV – теперішня вартість інвестицій, грн.;

$Т_ж$ – життєвий цикл наукової розробки, роки.

$$Е_в = \sqrt[3]{1 + \frac{16\,790\,715{,}6}{1\,322\,784{,}4}} - 1 = 1{,}39 \text{ або } 139\%$$

Проведемо порівняння значення $Е_в$ із мінімальною (бар'єрною) ставкою дисконтування τ_{min}, яка визначає найнижчий рівень доходності, нижче якого інвестиції не є доцільними.

У загальному висловленні мінімальна (бар'єрна) ставка дисконтування τ_{min} визначається за визначеною формулою:

$$\tau_{min} = d + f$$

(4.13)

d – середньозважена ставка за депозитними операціями в комерційних банка;

f – показник, що характеризує ризикованість вкладень; *f* = 0,2.

d = 0,12.

$$\tau_{min} = 0{,}2 + 0{,}12 = 0{,}32$$

Зважаючи на те, що $Е_в$ дорівнює 139%, що перевищує τ_{min} = 32%, можна зазначити, що інвестор виявляє потенційний інтерес до фінансування даної наукової розробки.

На шостому етапі проведемо розрахунок періоду окупності вкладених інвестицій у реалізацію наукового проекту ($Т_{ок}$) за визначеною формулою:

$$Т_{ок} = \frac{1}{Е_в}, \text{рік}$$

(4.14)

$$Т_{ок} = \frac{1}{1{,}39} = 0{,}72 \text{ року}$$

З урахуванням того, що термін окупності витрат, вкладених у впровадження наукового проекту, становить менше трьох років, можна зробити висновок, що фінансування нової розробки є не тільки доцільним, але й економічно обґрунтованим. Подальше інвестування у проект може призвести до стабільного зростання прибутковості та позитивно вплинути на фінансовий успіх підприємства.

4.5 Висновки до розділу

У даному розділі було проведено оцінювання комерційного потенціалу розробки програмного засобу для захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону. Технологічний аудит залучав участь трьох незалежних експертів, які визначили, що рівень комерційного потенціалу розробки перевищує середній рівень.

Відповідно до проведеного оцінювання, розробка виявилася якісною та конкурентоспроможною. Рівень комерційного потенціалу розробки становить 44, що відповідає рівню "високий". З розрахунків витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи видно, що загальні витрати на розробку складають 440 928,14 грн.. Розрахована абсолютна ефективність вкладених інвестицій в сумі 16 790 715,6 грн. свідчить про те, що інвестор отримає прибуток від комерціалізації програмного продукту.

Щорічна ефективність вкладених інвестицій в наукову розробку складає 139%, що перевищує мінімальну бар'єрну ставку дисконтування у 33%. Це свідчить про зацікавленість інвесторів у фінансуванні нової розробки. Термін окупності вкладених інвестицій у реалізацію проекту складає 0,72 ріку, що також підтверджує доцільність фінансування нової розробки.

Отже, аналізуючи отримані економічні показники, можна вважати, що запропонована розробка програмного засобу має високий комерційний потенціал і, отже, є доцільною для подальшого впровадження.

ВИСНОВКИ

У даній магістерській роботі проведено глибоке наукове дослідження та розробку "Захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону". Вибір цієї теми обумовлено актуальністю, враховуючи різко зростаючі виклики та потреби в галузі хімічної інфраструктури.

Швидкий темп технологічного розвитку та зростання обсягів виробничої інформації створюють серйозні проблеми у сфері кібербезпеки та конфіденційності даних у галузі хімічної інфраструктури. Зростання кількості цифрових атак та загроз безпеці вимагає ефективних заходів для захисту хімічних об'єктів та їх інфраструктури. Таким чином, розробка захищеного консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури стає надзвичайно важливою задачею, яка відповідає сучасним викликам і потребам галузі.

Основний акцент дослідження зосереджений на системному підході до забезпечення безпеки хімічної інфраструктури в конкретному регіоні за допомогою реалізації розробленого консолідованого інформаційного ресурсу. Робота включає аналіз теоретичних матеріалів, визначення особливостей критичної хімічної інфраструктури та встановлення принципів, які слід враховувати при розробці консолідованого інформаційного ресурсу.

Досліджено принципи та методи збору та обробки даних для системного аналізу, а також проаналізовано питання забезпечення безпеки хімічних об'єктів. Сформульовані висновки на основі проведеного аналізу та визначені ключові завдання для подальших етапів дослідження.

У другому розділі детально розглянуто створення "Захищеного консолідованого інформаційного ресурсу аналізу безпеки хімічної інфраструктури". В основу розділу покладено особливості розробки інформаційного ресурсу для аналізу безпеки хімічної інфраструктури, а також

розглянуті вимоги, необхідні для забезпечення ефективності та надійності ресурсу.

У рамках розділу виконано розробку бази даних консолідованого інформаційного ресурсу, використовуючи метод сутність-зв'язок. Визначено сутності та їх взаємозв'язки для оптимального зберігання та організації інформації про безпеку хімічної інфраструктури. Проведено нормалізацію відношень бази даних з метою поліпшення структури та уникнення аномалій.

У третьому розділі здійснено практичну реалізацію бази даних, враховуючи специфіку хімічної сфери, та проведено програмну реалізацію звітів з бази даних. Також виконано практичну реалізацію системи системного аналізу безпеки, аналізовано системну безпеку хімічної інфраструктури регіону на основі впроваджених програмних рішень.

Четвертий розділ роботи включає аналіз економічної доцільності розробки та впровадження програмного забезпечення, причому наведені економічні показники свідчать про високий комерційний потенціал розробленого продукту.

У підсумку, магістерська робота успішно досягла свого головного завдання, презентувавши розробку захищеного консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури регіону, та визначила напрямки подальшого вдосконалення цієї системи в майбутньому.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Cloudflare Application Services | Security and Performance. Cloudflare. URL: https://www.cloudflare.com/application-services/?utm_source=google&utm_medium=cpc&utm_campaign=DG_EMEA_ENG_G_Search_Generic_Beta_Applications-Security&utm_content=Beta_Generic_Applications-Security_Core&utm_term=cyber+security&campaignid=7170000011271632 2&adgroupid=58700008486001012&creativeid=662071359069&gclid=Cj0KCQiA67CrBhC1ARIsACKAa8Q17aOUR4pjkbG6p-6JfA6-aLhFR_-ig7BqZ7VtZrN6wUzk1Nhp6nkaAgwEEALw_wcB&gclsrc=aw.ds).

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

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

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

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

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

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

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

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

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

11. Wright G. What is critical infrastructure? | Definition from TechTarget. WhatIs.com. URL: https://www.techtarget.com/whatis/definition/critical-infrastructure#:~:text=Critical%20infrastructure%20is%20the%20collection,each%20nation%20considers%20critical%20varies.

12. Critical infrastructure sectors | CISA. Cybersecurity and Infrastructure Security Agency CISA. URL: https://www.cisa.gov/topics/critical-infrastructure-security-and-resilience/critical-infrastructure-sectors.

13. Contributors to Wikimedia projects. Critical infrastructure - Wikipedia. Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Critical_infrastructure.

14. Critical infrastructure. Migration and Home Affairs. URL: https://home-affairs.ec.europa.eu/pages/page/critical-infrastructure_en.

15. Critical infrastructure security and resilience | cybersecurity and infrastructure security agency CISA. Home Page | CISA. URL: https://www.cisa.gov/topics/critical-infrastructure-security-and-resilience.

16. Critical infrastructure | homeland security. Home | Homeland Security. URL: https://www.dhs.gov/science-and-technology/critical-infrastructure.

17. Five things you need to know about critical infrastructures - Institute for Environment and Human Security. Institute for Environment and Human Security. URL: https://ehs.unu.edu/blog/5-facts/5-things-about-critical-infrastructures.html.

18. What is critical infrastructure?. BlackBerry. Intelligent Security. Everywhere. URL: https://www.blackberry.com/us/en/solutions/endpoint-security/industry-4-0/critical-infrastructure.

19. Cyber and infrastructure security centre website. Cyber and Infrastructure Security Centre Website. URL: https://www.cisc.gov.au/what-is-the-cyber-and-infrastructure-security-centre/defining-critical-infrastructure.

20. Critical infrastructure protection. EU Science Hub. URL: https://joint-research-centre.ec.europa.eu/scientific-activities-z/critical-infrastructure-protection_en.

21. Critical infrastructure - Glossary | CSRC. NIST Computer Security Resource Center | CSRC. URL: https://csrc.nist.gov/glossary/term/critical_infrastructure.

22. Critical infrastructure. Cyberwatching. URL: https://www.cyberwatching.eu/cybersecurity-and-privacy-project-clusters/critical-infrastructure.

23. What is critical infrastructure? Why does critical infrastructure security matter?. Palo Alto Networks. URL: https://www.paloaltonetworks.com/cyberpedia/what-is-critical-infrastructure.

24. Critical national infrastructure | NPSA. National Protective Security Authority | NPSA. URL: https://www.npsa.gov.uk/critical-national-infrastructure-0.

25. What is critical infrastructure? | IBM. IBM in Deutschland, Österreich und der Schweiz | IBM. URL: https://www.ibm.com/topics/critical-infrastructure.

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

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

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

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

30. What is blockchain?. *McKinsey & Company*. URL: https://www.mckinsey.com/featured-insights/mckinsey-explainers/what-is-blockchain .

31. How does it work? | built in. URL: https://builtin.com/132435.

32. What is blockchain technology - IBM blockchain | IBM. *IBM - Deutschland | IBM*. URL: https://www.ibm.com/topics/blockchain .

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

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

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

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

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

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

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

40. 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/

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

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

43. 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
"20" вересня 2023 р.

ТЕХНІЧНЕ ЗАВДАННЯ

до магістерської кваліфікаційної роботи на тему:
Захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону
08-72.МКР.009.00.000.ТЗ

Керівник магістерської кваліфікаційної роботи
к т.н., проф. каф. МБІС
Азарова А.О.

Вінниця – 2023 р.

1. Найменування та область застосування

Захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону

2.Підстава для розробки

Розробка виконується на основі наказу ректора ВНТУ № 247 від 18 вересня 2023 р.

3. Мета та призначення розробки

3.1 Мета розробки: розробка захищеного консолідованого інформаційного ресурсу системного аналізу безпеки хімічної інфраструктури регіону.

3.2 Призначення: системний аналіз безпеки хімічної інфраструктури та захист цих даних.

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 методу 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	20.09.2023	31.09.2023	
2.	Аналіз предметної області обраної теми	01.10.2023	15.10.2023	
3.	Розробка роботи	16.10.2023	26.10.2023	
4.	Написання магістерської роботи на основі розробленої теми	27.10.2023	15.11.2023	
5.	Передзахист магістерської кваліфікаційної роботи	16.11.2023	24.11.2023	
6.	Виправлення, уточнення, корегування магістерської кваліфікаційної роботи	27.11.2023	04.12.2023	
7.	Захист магістерської кваліфікаційної роботи	11.12.2023	17.12.2023	

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 _____Скомаровський В.В

Додаток Б. Лістинг програми

```
import { HttpStatus, Inject, Injectable } from '@nestjs/common';
import { QuizzesModel } from './entities/quizzes.model';
import { DeleteResult, Repository } from 'typeorm';
import { InjectRepository } from '@nestjs/typeorm';
import { JWTPayload } from 'src/auth/interfaces/jwtAuth0.payload.interface';
import { CompanyModel } from 'src/companies/entiites/companies.model';
import { MembersModel, Roles } from 'src/companies/entiites/members.model';
import { GeneralResponse } from 'src/dto/responseTemplate.dto';
import { UsersService } from 'src/users/users.service';
import { QuizDto } from './dto/quizzes.dto';
import { PaginateQuery, Paginated, paginate } from 'nestjs-paginate';
import { PassedQuestiosnDto } from './dto/passedQuestions.dto';
import { ResultOfQuizForRedis } from './interfaces/pasedQyizResultForRedis.interface';
import { MemberRatingModel } from './entities/memberRating.model';
import { UserRatingModel } from './entities/userRating.model';
import { QuizzesResultsModel } from './entities/quizzesResults.model';
import { DatesOfLastQuizzesPassedModel } from './entities/datesOfLastQuizeesPased.model';
import { CACHE_MANAGER } from '@nestjs/cache-manager';
import { Cache } from 'cache-manager';

@Injectable()
export class QuizzesService {
 constructor(
  @InjectRepository(QuizzesModel)
  private quizzesRepository: Repository<QuizzesModel>,
  @InjectRepository(CompanyModel)
  private companyRepository: Repository<CompanyModel>,
  @InjectRepository(MembersModel)
  private membersRepository: Repository<MembersModel>,
  @InjectRepository(MemberRatingModel)
  private membersRatingRepository: Repository<MemberRatingModel>,
  @InjectRepository(UserRatingModel)
  private userRatingRepository: Repository<UserRatingModel>,
  @InjectRepository(QuizzesResultsModel)
  private quizzesResultsRepository: Repository<QuizzesResultsModel>,
  @InjectRepository(DatesOfLastQuizzesPassedModel)
  private dateOfLastQuizzesPassedRepository: Repository<DatesOfLastQuizzesPassedModel>,
  private userService: UsersService,
  @Inject(CACHE_MANAGER) private cacheService: Cache,
 ) { }

 private async isCompanyAdminOrOwner(
  email: string,
  companyId: number,
 ): Promise<boolean> {
  const company = await this.companyRepository.findOne({
   where: { id: companyId },
  });
  if (!company) {
   return false;
  }
  const user = await this.userService.getUserByEmail(email);
  if (!user) {
   return false;
  }
  const member = await this.membersRepository.findOne({
```

```
    where: { userId: user.id, companyId: companyId, role: Roles.ADMIN },
   });
   if (member || company.ownerId === user.id) {
    return true;
   } else {
    return false;
   }
  }

  public async addNewQuiz(
   payload: JWTPayload,
   companyId: number,
   dto: QuizDto,
  ): Promise<GeneralResponse<QuizzesModel>> {
   try {
    const company = await this.companyRepository.findOne({
     where: { id: companyId },
    });
    if (!company) {
     throw new Error('Company not found');
    }
    const isAdminOrOwner = await this.isCompanyAdminOrOwner(
     payload.email,
     companyId,
    );
    if (!isAdminOrOwner) {
     throw new Error('Quiz can create only by company owner or admin');
    }

    const newQuiz = await this.quizzesRepository.create({
     ...dto,
     companyId: companyId,
    });
    await this.quizzesRepository.save(newQuiz);
    return new GeneralResponse(
     newQuiz,
     'Quiz is created',
     HttpStatus.CREATED,
    );
   } catch ({ error, message }) {
    return new GeneralResponse(
     message,
     'Quiz not created',
     HttpStatus.BAD_REQUEST,
    );
   }
  }

  public async deleteQuiz(
   payload: JWTPayload,
   quizId: number,
  ): Promise<GeneralResponse<DeleteResult>> {
   try {
    const quiz = await this.quizzesRepository.findOne({
     where: { id: quizId },
    });
    if (!quiz) {
     throw new Error('Quiz not found');
```

```
    }
    const company = await this.companyRepository.findOne({
      where: { id: quiz.companyId },
    });
    if (!company) {
      throw new Error('Company not found');
    }
    const isAdminOrOwner = await this.isCompanyAdminOrOwner(
      payload.email,
      quiz.companyId,
    );
    if (!isAdminOrOwner) {
      throw new Error('Quiz can delete only by company owner or admin');
    }
    return new GeneralResponse(
      await this.quizzesRepository.delete(quizId),
      'Quiz is deleted',
      HttpStatus.OK,
    );
  } catch ({ error, message }) {
    return new GeneralResponse(
      message,
      'Quiz not deleted',
      HttpStatus.BAD_REQUEST,
    );
  }
}

public async updateQuiz(
  payload: JWTPayload,
  quizId: number,
  dto: QuizDto,
): Promise<GeneralResponse<QuizzesModel>> {
  try {
    const quiz = await this.quizzesRepository.findOne({
      where: { id: quizId },
    });
    if (!quiz) {
      throw new Error('Quiz not found');
    }
    const isAdminOrOwner = await this.isCompanyAdminOrOwner(
      payload.email,
      quiz.companyId,
    );
    if (!isAdminOrOwner) {
      throw new Error('Quiz can delete only by company owner or admin');
    }
    await this.quizzesRepository.update(quizId, dto);
    return new GeneralResponse(
      await this.quizzesRepository.findOne({ where: { id: quizId } }),
      'Quiz is updated',
      HttpStatus.OK,
    );
  } catch ({ error, message }) {
    return new GeneralResponse(
      message,
      'Quiz is not updated',
      HttpStatus.BAD_REQUEST,
```

```
    );
  }
}

public async getQuizById(
  quizId: number,
): Promise<GeneralResponse<QuizzesModel>> {
  try {
    const quiz = await this.quizzesRepository.findOne({
      where: { id: quizId },
    });
    if (!quiz) {
      throw new Error('Quiz not found');
    }
    return new GeneralResponse(quiz, 'Get quiz by id', HttpStatus.OK);
  } catch ({ error, message }) {
    return new GeneralResponse(
      message,
      'Quiz is not updated',
      HttpStatus.BAD_REQUEST,
    );
  }
}

public async allCompanyQuizzes(
  query: PaginateQuery,
  companyId: number,
): Promise<GeneralResponse<Paginated<QuizzesModel>>> {
  const paginateQuizzes = paginate(query, this.quizzesRepository, {
    sortableColumns: [
      'id',
      'quizName',
      'quizDescription',
      'created_at',
      'updated_at',
    ],
    nullSort: 'last',
    defaultSortBy: [['id', 'DESC']],
    searchableColumns: ['quizName', 'quizDescription', 'id'],
    select: [
      'id',
      'quizName',
      'quizDescription',
      'questions',
      'frequency',
      'companyId',
      'created_at',
      'updated_at',
    ],
    where: { companyId: companyId },
  });
  return new GeneralResponse(
    await paginateQuizzes,
    'Quizzes list',
    HttpStatus.OK,
  );
}
```

```
public async passQuiz(
  payload: JWTPayload,
  quizId: number,
  passedQuestions: PassedQuestiosnDto[],
): Promise<GeneralResponse<QuizzesResultsModel>> {
  try {
    const quiz = await this.quizzesRepository.findOne({
      where: { id: quizId },
    });
    if (!quiz) {
      throw new Error('Quiz not found');
    }
    const user = await this.userService.getUserByEmail(payload.email);
    if (!user) {
      throw new Error('User is not found');
    }
    const company = await this.companyRepository.findOne({
      where: { id: quiz.companyId },
    });
    if (!company) {
      throw new Error('Company not found');
    }
    const member = await this.membersRepository.findOne({
      where: { userId: user.id, companyId: company.id },
    });
    if (!member) {
      throw new Error('Member not found');
    }
    const resultOfQuizPassed: {
      numberOfAllQuestin: number;
      numberOfCorectAnswers: number;
    } = {
      numberOfAllQuestin: quiz.questions.length,
      numberOfCorectAnswers: 0,
    };
    if (quiz.questions.length !== passedQuestions.length) {
      throw new Error('The number of questions does not match');
    }

    //redis data
    const resultForRedis: ResultOfQuizForRedis = {
      memberId: member.id,
      companyId: company.id,
      quizId: quiz.id,
      passedQuestionsResult: [],
    };


    //checking the results of the quiz

    passedQuestions.map((value, index) => {
      const question = quiz.questions[index];
      const compareAnsvers = question.corectAnswer.reduce(
        (score: number, element: number) =>
          score + Number(value.answers.includes(element)),
        0,
      );
      const points = compareAnsvers / question.corectAnswer.length;
```

```
  const fines =
    value.answers.filter(
      (element) => !question.corectAnswer.includes(element),
    ).length / question.corectAnswer.length;
  const totalPoints = points - fines;
  if (totalPoints > 0) {
    resultOfQuizPassed.numberOfCorectAnswers += totalPoints;
  }
  resultForRedis.passedQuestionsResult.push({
    question: value.question,
    answers: value.answers,
    numberCorectAnswers:
      totalPoints > 0 ? Number(totalPoints.toFixed(3)) : 0,
  });
});

// update member rating
const memberRating = await this.membersRatingRepository.findOne({
  where: { memberId: member.id },
});
if (!memberRating) {
  const averageScore = Number(
    (
      resultOfQuizPassed.numberOfCorectAnswers /
      resultOfQuizPassed.numberOfAllQuestin
    ).toFixed(3),
  );
  const newMemberRating = await this.membersRatingRepository.create({
    memberId: member.id,
    allCorrectlyPassedQuestions: Number(
      resultOfQuizPassed.numberOfCorectAnswers.toFixed(3),
    ),
    allPassedQuestions: resultOfQuizPassed.numberOfAllQuestin,
    averageScore: averageScore,
  });
  await this.membersRatingRepository.save(newMemberRating);
} else {
  const updateMemberAllCorrectlyPassedQuestions = Number(
    (
      Number(memberRating.allCorrectlyPassedQuestions) +
      resultOfQuizPassed.numberOfCorectAnswers
    ).toFixed(3),
  );
  const updateMemberAllPassedQuestions =
    memberRating.allPassedQuestions +
    resultOfQuizPassed.numberOfAllQuestin;
  const updateMemberAverageScore = Number(
    (
      updateMemberAllCorrectlyPassedQuestions /
      updateMemberAllPassedQuestions
    ).toFixed(3),
  );
  await this.membersRatingRepository.update(memberRating.id, {
    allCorrectlyPassedQuestions: updateMemberAllCorrectlyPassedQuestions,
    allPassedQuestions: updateMemberAllPassedQuestions,
    averageScore: updateMemberAverageScore,
  });
}
```

```
//update user rating
const userRating = await this.userRatingRepository.findOne({
  where: { userId: user.id },
});
if (!userRating) {
  const averageScore = Number(
    (
      resultOfQuizPassed.numberOfCorectAnswers /
      resultOfQuizPassed.numberOfAllQuestin
    ).toFixed(3),
  );
  const newUserRating = await this.userRatingRepository.create({
    userId: user.id,
    allCorrectlyPassedQuestions: Number(
      resultOfQuizPassed.numberOfCorectAnswers.toFixed(3),
    ),
    allPassedQuestions: resultOfQuizPassed.numberOfAllQuestin,
    averageScore: averageScore,
  });
  await this.userRatingRepository.save(newUserRating);
} else {
  const updateUserRatingAllCorrectlyPassedQuestions = Number(
    (
      Number(userRating.allCorrectlyPassedQuestions) +
      resultOfQuizPassed.numberOfCorectAnswers
    ).toFixed(3),
  );
  const updateUserRatingAllPassedQuestions =
    userRating.allPassedQuestions + resultOfQuizPassed.numberOfAllQuestin;
  const updateUserRatingAverageScore = Number(
    (
      updateUserRatingAllCorrectlyPassedQuestions /
      updateUserRatingAllPassedQuestions
    ).toFixed(3),
  );
  await this.userRatingRepository.update(userRating.id, {
    allCorrectlyPassedQuestions:
      updateUserRatingAllCorrectlyPassedQuestions,
    allPassedQuestions: updateUserRatingAllPassedQuestions,
    averageScore: updateUserRatingAverageScore,
  });
}

//update date of last quiz passed
const dateOfLastQuizPassed =
  await this.dateOfLastQuizzesPassedRepository.findOne({
    where: { memberId: member.id, quizId: quiz.id },
  });
if (!dateOfLastQuizPassed) {
  const newDateOfLastQuizPassed =
    await this.dateOfLastQuizzesPassedRepository.create({
      memberId: member.id,
      quizId: quiz.id,
      dateOfQuizLastPassed: new Date(),
    });
  await this.dateOfLastQuizzesPassedRepository.save(
    newDateOfLastQuizPassed,
  );
```

```
    } else {
      await this.dateOfLastQuizzesPassedRepository.update(
        dateOfLastQuizPassed.id,
        { dateOfQuizLastPassed: new Date() },
      );
    }

    //save quiz result
    const quizResult = await this.quizzesResultsRepository.create({
      quizId: quiz.id,
      memberId: member.id,
      correctlyPassedQuestions: Number(
        resultOfQuizPassed.numberOfCorectAnswers.toFixed(3),
      ),
      passedQuestions: resultOfQuizPassed.numberOfAllQuestin,
    });
    await this.quizzesResultsRepository.save(quizResult);

    //save data to redis
    await this.cacheService.set(quizResult.id.toString(), resultForRedis, 60 * 60 * 24 * 2);

    return new GeneralResponse(quizResult, 'Quiz is passed', HttpStatus.OK);
   } catch ({ error, message }) {
    return new GeneralResponse(
      message,
      'Quiz is not passed',
      HttpStatus.BAD_REQUEST,
    );
   }
  }
}
import {
  ForbiddenException,
  HttpStatus,
  Injectable,
  UnauthorizedException,
} from '@nestjs/common';
import { CreateUserDto } from 'src/users/dto/createUser.dto';
import { UsersService } from 'src/users/users.service';
import { LoginDto } from './dto/userLogin.dto';
import { comparePassword } from 'src/utils/bcrypt.utils';
import { UserModel } from 'src/users/user.model';
import { JwtService } from '@nestjs/jwt';
import { GeneralResponse } from 'src/dto/responseTemplate.dto';
import { UserDto } from 'src/users/dto/user.dto';
import { v4 as uuidv4 } from 'uuid';
import { GenerateTokensDto } from './dto/generateTokens.dto';
import { EmailService } from 'src/email/email.service';
@Injectable()
export class AuthService {
  constructor(
    private usersService: UsersService,
    private jwtService: JwtService,
    private emailService: EmailService,
  ) {}

  public async login(
    userDto: LoginDto,
```

```
): Promise<GeneralResponse<GenerateTokensDto>> {
  try {
    const user = await this.validateUser(userDto);
    if (!user) {
      throw new Error('wrong email or password');
    }
    return new GeneralResponse(
      await this.generateTokens(user),
      'Authorization complete',
      HttpStatus.OK,
    );
  } catch ({ error, message }) {
    return new GeneralResponse(
      message,
      'Authorization not complete',
      HttpStatus.BAD_REQUEST,
    );
  }
}

public async registration(
  userDto: CreateUserDto,
): Promise<GeneralResponse<UserDto>> {
  return this.usersService.createUser(userDto);
}

public async getMe(
  payload,
): Promise<
  GeneralResponse<UserModel> | GeneralResponse<GeneralResponse<UserModel>>
> {
  const user = await this.usersService.getUserByEmail(payload.email);
  if (user) {
    return new GeneralResponse(user, 'Get me info', HttpStatus.OK);
  }
  const newAuth0User = await this.usersService.createUser({
    userName: payload.email,
    email: payload.email,
    password: await uuidv4(),
  });
  return new GeneralResponse(newAuth0User, 'Get me info', HttpStatus.OK);
}

public async refreshTokens(body: {
  token: string;
}): Promise<GeneralResponse<GenerateTokensDto>> {
  try {
    const payload = await this.jwtService.verifyAsync(body.token, {
      secret: process.env.JWT_REFRESH_SECRET,
    });
    const user = await this.usersService.getUserByEmail(payload.email);
    if (!user) {
      throw new ForbiddenException('Access Denied');
    }
    return new GeneralResponse(
      await this.generateTokens(user),
      'Tokens refreshed',
      HttpStatus.OK,
```

```
    );
  } catch (error) {
    throw new UnauthorizedException();
  }
}

private async generateTokens(user: UserModel): Promise<GenerateTokensDto> {
  const payload = {
    email: user.email,
  };

  const [accessToken, refreshToken, actionToken] = await Promise.all([
    this.jwtService.signAsync(payload, {
      secret: process.env.JWT_KEY,
      expiresIn: '15m',
    }),
    this.jwtService.signAsync(payload, {
      secret: process.env.JWT_REFRESH_SECRET,
      expiresIn: '7d',
    }),
    this.jwtService.signAsync(payload, {
      secret: process.env.JWT_ACTION_SECRET,
      expiresIn: '7d',
    }),
  ]);
  return {
    accessToken,
    refreshToken,
    actionToken,
  };
}

private async validateUser(userDto: LoginDto): Promise<UserModel> {
  const user = await this.usersService.getUserByEmail(userDto.email);
  const passwordEquals = await comparePassword(
    userDto.password,
    user.password,
  );
  if (user && passwordEquals) {
    return user;
  }
  return null;
}

public async sendEmailForgotPassword(
  email: string,
): Promise<GeneralResponse<string>> {
  try {
    const user = await this.usersService.getUserByEmail(email);
    if (!user) {
      throw new Error('User not found');
    }
    const token = await this.jwtService.signAsync(
      { email: email },
      {
        secret: process.env.EMAIL_TOKEN_SECRET,
        expiresIn: '15m',
      },
```

```
      );
      await this.emailService.sendEmailForgotPassword(token, email);
      return new GeneralResponse('Done', 'Email send', HttpStatus.OK);
    } catch ({ error, message }) {
      return new GeneralResponse(
        message,
        'Email not send',
        HttpStatus.BAD_REQUEST,
      );
    }
  }

  public async changePassword(
    token: string,
    password: string,
  ): Promise<GeneralResponse<string>> {
    try {
      const payload: { email: string } = await this.jwtService.verifyAsync(
        token,
        { secret: process.env.EMAIL_TOKEN_SECRET },
      );
      if (payload) {
        await this.usersService.changePassword(payload.email, password);
        return new GeneralResponse('Done', 'Password chenged', HttpStatus.OK);
      } else {
        throw new Error('Bad token');
      }
    } catch ({ error, message }) {
      return new GeneralResponse(
        message,
        'Password not chenged',
        HttpStatus.BAD_REQUEST,
      );
    }
  }
}
```

Додаток В. Ілюстративний матеріал

ЗАХИЩЕНИЙ КОНСОЛІДОВАНИЙ ІНФОРМАЦІЙНИЙ РЕСУРС СИСТЕМНОГО АНАЛІЗУ БЕЗПЕКИ ХІМІЧНОЇ ІНФРАСТРУКТУРИ РЕГІОНУ

ВИКОНАВ: СТУДЕНТ ГРУПИ КІТС – 22М СКОМАРОВСЬКИЙ В.В

КЕРІВНИК: , К Т.Н., ПРОФ. КАФ. МБІС АЗАРОВА А.О.

ВСТУП

- Хімічна інфраструктура є вразливою до атак з різних причин. Зловмисники можуть атакувати хімічну інфраструктуру для отримання економічної вигоди, політичної мети або просто для завдавання шкоди.
- Атаки на хімічну інфраструктуру можуть мати масштабні наслідки. Вони можуть призвести до забруднення навколишнього середовища, шкоди здоров'ю людей та навіть до людських жертв.
- Тому було вирішено розробити захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону, який буде не лише аналізувати безпеку хімічної інфраструктури, а й захищати отримані дані

АКТУАЛЬНІСТЬ ТА МЕТА

Актуальність обраної теми надзвичайно важлива в контексті стрімкого розвитку хімічної галузі та супутнього зростання ризиків, пов'язаних із безпекою виробничих об'єктів. А й також зв'язана з розробкою захищеного консолідованого інформаційного ресурсу

Мета даного дослідження полягає в розробці ресурсу, спрямованого на забезпечення аналізу безпеки хімічної інфраструктури регіону та забезпечення безпеки даних та безпеки самого інформаційного ресурсу.

КРИТИЧНА ІНФРАСТРУКТУРА В ХІМІЧНІЙ ГАЛУЗІ

Хімічна галузь

- Хімічна галузь визначається як критична інфраструктура, яка має стратегічне значення для економічного розвитку та забезпечення потреб національної безпеки. Однак, разом з великим потенціалом для інновацій та економічного зростання, хімічна інфраструктура також стикається з унікальними викликами безпеки.

Основні характеристики

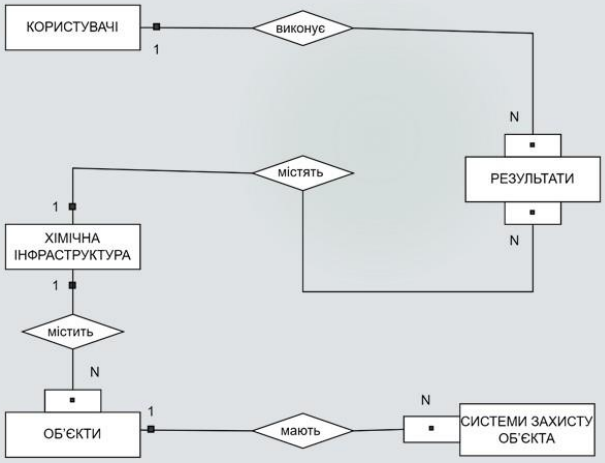
- Основні характеристики, що роблять хімічну інфраструктуру критичною, включають велику комплексність технологічних процесів, обсяги небезпечних речовин, та високий рівень потенційної небезпеки при неналежному управлінні.

Розробка

- Створення захищеного консолідованого інформаційного ресурсу в умовах хімічної інфраструктури має на меті виявлення, аналіз та покращення систем безпеки, сприяючи стабільності та розвитку цієї важливої галузі.

Розробка бази даних

На етапі визначення вимог для розробки бази даних консолідованого інформаційного ресурсу аналізу безпеки хімічної інфраструктури важливо було провести аналіз потреб у даних. В результаті аналізу була розроблена наступна ER-модель бази даних,

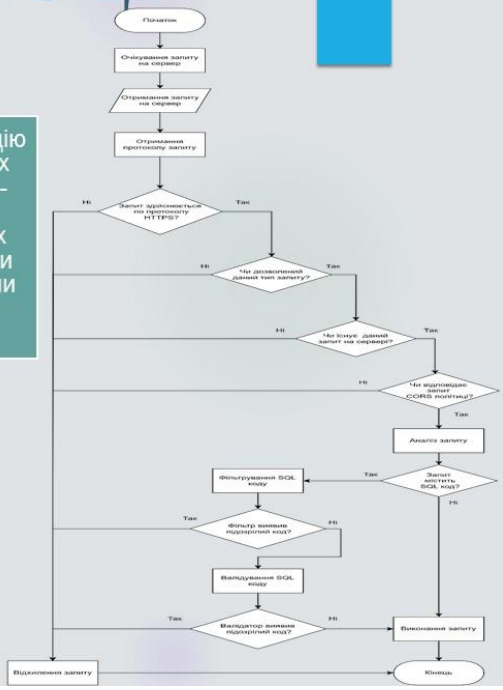


АЛГОРИТМ ЗАХИСТУ ВІД SQL-ІН'ЄКЦІЙ

При розробці безпеки від SQL-ін'єкцій важливо використовувати параметризовані запити для уникнення введення SQL-коду, що може призвести до небажаних операцій або розголошення конфіденційної інформації.

Це можна досягти через валідацію та екранування введених даних перед використанням їх у SQL-запитах. Використання параметризованих збережених процедур також може зменшити ризик SQL-ін'єкцій, оскільки вони можуть бути підготовлені та скомпільовані заздалегідь.

Крім того, важливо обмежувати привілеї користувачів, надаючи їм лише необхідні для завдань привілеї, тим самим зменшуючи можливість використання SQL-ін'єкцій для несанкціонованого доступу.



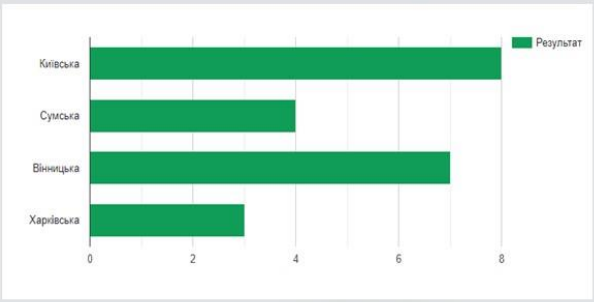
Використані технології

koa

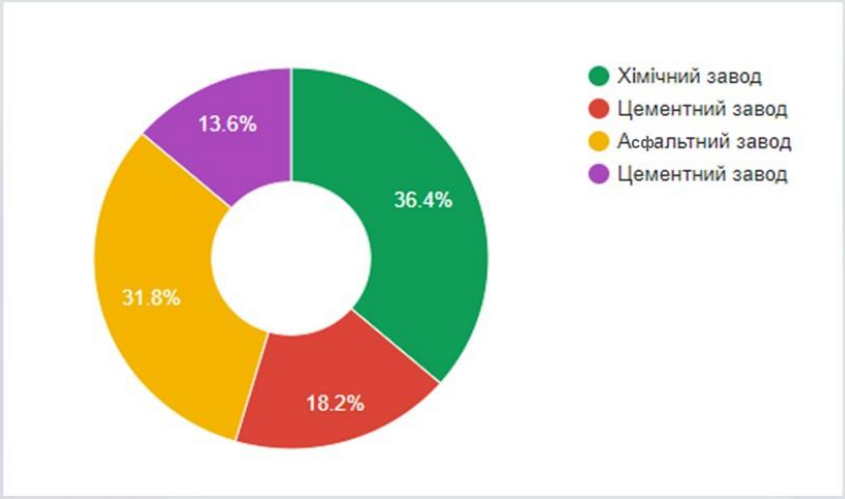
React

PostgreSQL

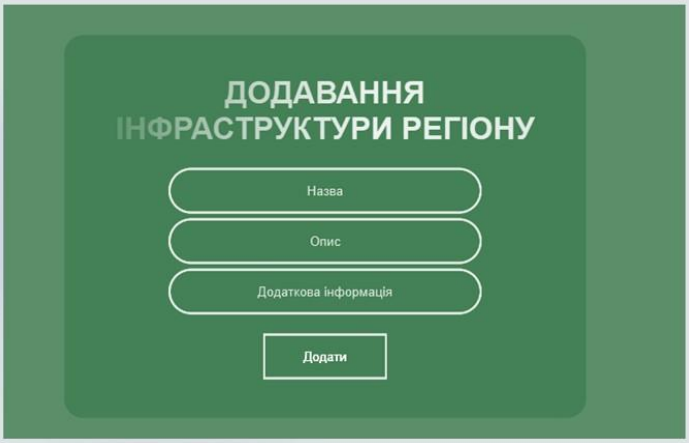
РЕАЛІЗАЦІЯ ЗАПИТІВ ТА ЗВІТІВ

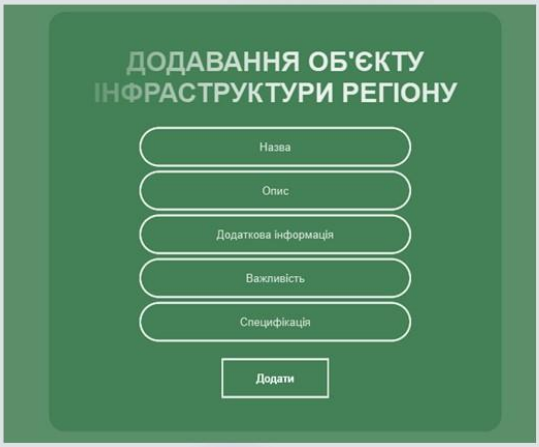


РЕАЛІЗАЦІЯ ЗАПИТІВ ТА ЗВІТІВ

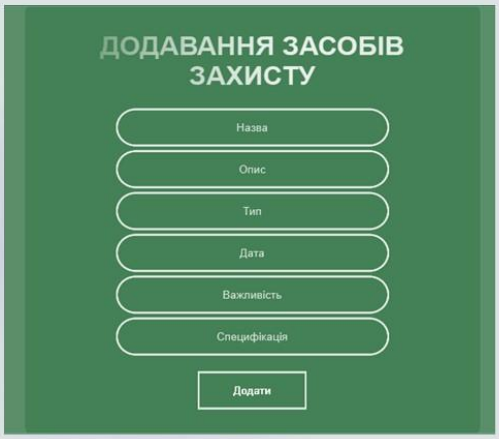


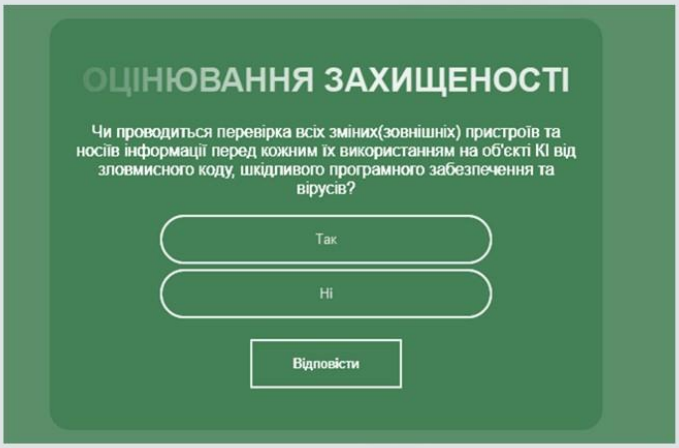
ІНТЕРФЕЙС РОЗРОБКИ





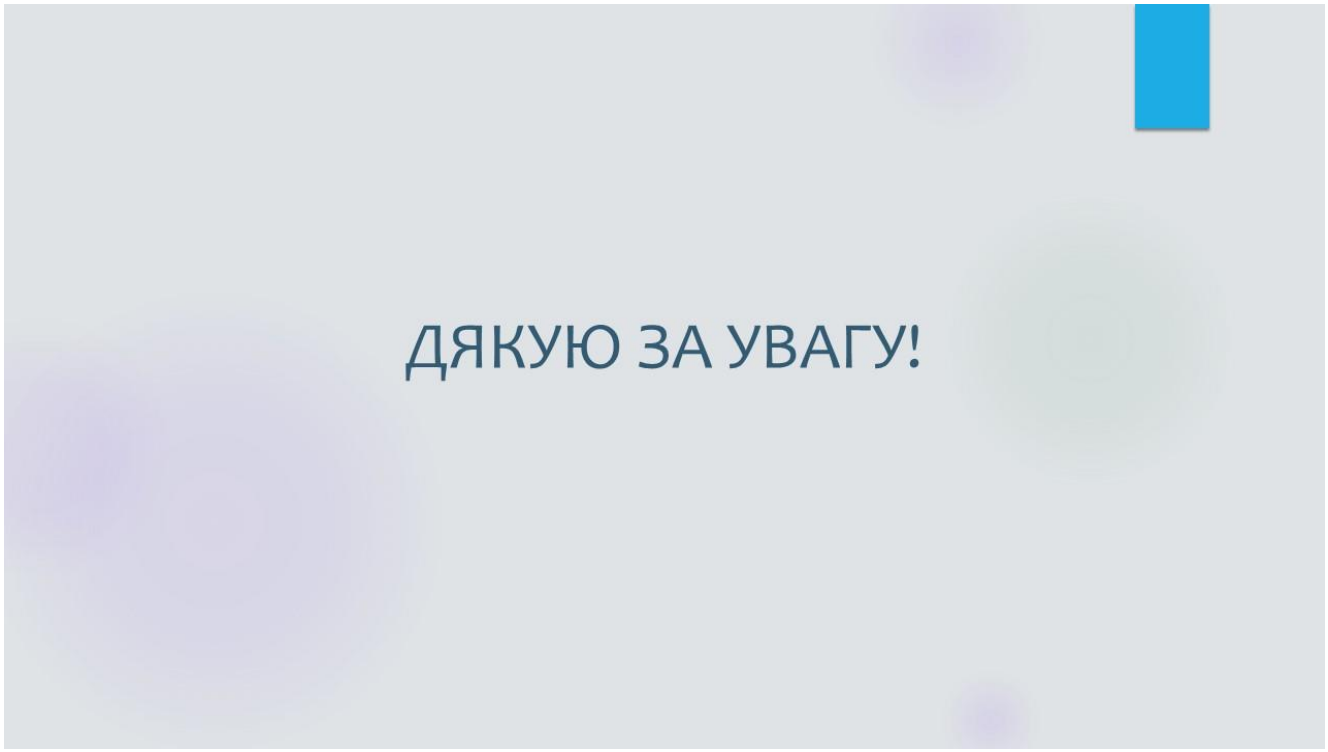
ІНТЕРФЕЙС РОЗРОБКИ





ВИСНОВКИ

- В результаті проведеного наукового дослідження та розробки захищеного консолідованого інформаційного ресурсу для системного аналізу безпеки хімічної інфраструктури регіону вдається глибоко проаналізувати особливості розробки захищених інформаційних ресурсів та врахувати унікальні особливості цієї критичної галузі .
- За допомогою розробленого інформаційного ресурсу вдалося визначити та вирішити ключові аспекти безпеки, спрямовані на забезпечення стійкості та надійності хімічної інфраструктури. Використання системного підходу та врахування специфіки хімічних процесів стали важливими факторами в досягненні цієї мети.
- Зазначене дослідження відкриває нові можливості для ефективного моніторингу, аналізу та підвищення рівня безпеки в хімічній галузі, сприяючи сталому розвитку та захисту цієї важливої галузі.


ДЯКУЮ ЗА УВАГУ!

Додаток Г. Протокол перевірки на антиплагіат

ПРОТОКОЛ
ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ
ЗАПОЗИЧЕНЬ

Назва роботи: Захищений консолідований інформаційний ресурс системного аналізу безпеки хімічної інфраструктури регіону
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 97 % Схожість 3 %

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 _____ Скомаровський В.В.
(підпис) (прізвище, ініціали)

Керівник роботи _____ Азарова А.О.
(підпис) (прізвище, ініціали)