Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації

Магістерська кваліфікаційна робота на тему:
«Система пошуку та аналізу небезпечного контенту інформаційних ресурсів»

Виконав: студент 2 курсу групи 2БС-22м
спеціальності 125 Кібербезпека
_____Руслан П'ЯТАК

Керівник: к. т. н., доцент каф. ЗІ
_____Олеся ВОЙТОВИЧ
«_12_» ____12_____2023 р.

Опонент: к. т. н., доцент каф. ПЗ
_____Олександр ХОШАБА
«_13_» ____12_____2023 р.

Допущено до захисту
Завідувач кафедри ЗІ
д. т. н., проф.
_____Володимир ЛУЖЕЦЬКИЙ
«_14_» _____12_____2023 р.

Вінниця 2023

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації
Рівень вищої освіти II (магістерський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 125 Кібербезпека
Освітньо-професійна програма – Безпека інформаційних і комунікаційних систем

ЗАТВЕРДЖУЮ
Завідувач кафедри ЗІ,
д. т. н., проф.
_____Володимир ЛУЖЕЦЬКИЙ
«__19__» ____09_____2023 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

П'ятаку Руслану Олеговичу

1. Тема роботи: «Система пошуку та аналізу небезпечного контенту інформаційних ресурсів»
керівник роботи: Войтович Олеся Петрівна, к. т. н., доцент кафедри ЗІ,
затверджені наказом ректора ВНТУ №247 від 18.09.2023р.

2. Строк подання студентом роботи: 13 грудня 2023року.

3. Вихідні дані до роботи:
- небезпечний контент;
- мова програмування JavaScript;
- тип програми – веб-додаток.

4. Зміст текстової частини: Вступ. 1. Аналіз методів пошуку та аналізу небезпечного контенту інформаційних ресурсів. 2. Метод пошуку та аналізу небезпечного контенту інформаційних ресурсів 3. Розробка структурної схеми системи 4. Розробка програмного засобу. 5. Економічна частина. Висновки. Перелік використаних джерел. Додатки.

5. Перелік ілюстративного матеріалу: Узагальнена архітектура системи (плакат А4). Метод пошуку небезпечного контенту (плакат А4). Метод аналізу(плакат А4). Архітектура засобу (плакат А4). Архітектура модуля пошуку (плакат А4). Алгоритм роботи засобу (плакат А4). Результати тестування (плакат А4).

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Олеся ВОЙТОВИЧ, к. т. н., доц. каф. ЗІ	19.09	26.09
2	Олеся ВОЙТОВИЧ, к. т. н., доц. каф. ЗІ	19.09	28.09
3	Олеся ВОЙТОВИЧ, к. т. н., доц. каф. ЗІ	19.09	21.10
4	Олеся ВОЙТОВИЧ, к. т. н., доц. каф. ЗІ	19.09	24.10
5	Ольга РАТУШНЯК, к.т.н., доц.каф ЕПВМ	19.09	24.10

7. Дата видачі завдання 1вересня 2023р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз завдання. Вступ	01.09.2023 – 10.09.2023	
2	Аналіз літературних джерел за напрямком магістерської роботи	10.09.2023 – 15.09.2023	
3	Розробка рішень	16.09.2023 – 22.09.2023	
4	Розробка модуля програмного засобу	30.09.2023 – 12.10.2023	
5	Практична реалізація, моделювання, експериментування, результати	14.10.2023 – 10.11.2023	
6	Розробка розділу тестування і обґрунтування доцільності розробки	11.11.2023 – 17.11.2023	
7	Аналіз виконання, висновки	18.11.2023 – 24.11.2023	
8	Оформлення пояснювальної записки	25.11.2023 – 30.11.2023	
9	Попередній захист та доопрацювання МКР	28.11.2023 – 10.12.2023	
10	Представлення МКР до захисту	11.12.2023 – 14.12.2023	
11	Захист МКР	14.12.2023 – 21.12.2023	

Студент _____Руслан П’ЯТАК

Керівник роботи _____Олеся ВОЙТОВИЧ

АНОТАЦІЯ

Магістерська кваліфікаційна робота складається з 85 сторінок формату А4, на яких є 34 рисунка, 6 таблиць, 19 формул, список використаних джерел містить 29 найменувань.

Магістерська кваліфікаційна робота присвячена розробці системи пошуку та аналізу небезпечного контенту інформаційних ресурсів. У роботі проаналізовано сучасні засоби захисту від небезпечного контенту, а також основних методів його пошуку та аналізу. У межах роботи виконано теоретико-множинний опис методу пошуку, описано метод аналізу та градацію класифікації небезпечного контенту. Спроектовано архітектуру системи, а також описано основні алгоритми роботи. Після розробки схем функціонування програмного засобу в цілому і алгоритмів його окремих складових здійснено програмну реалізацію. Проведено тестування системи на коректність роботи в реальних умовах. Проведено економічні розрахунки доцільності розробки.

Ключові слова: небезпечний контент, інформаційні ресурси, система, пошук, аналіз.

ABSTRACT

The master's thesis consists of 85 pages of A4 format, on which there are 34 figures, 6 tables, 19 formulas, the list of used sources contains 29 titles.

The master's thesis is devoted to the development of a system for searching and analyzing dangerous content of information resources. The work analyzes modern means of protection against dangerous content, as well as the main methods of its search and analysis. Within the framework of the work, a theoretical-multiple description of the search method was performed, the analysis method and the gradation of the classification of dangerous content were described. The system architecture is designed, and the main work algorithms are also described. After the development of schemes for the functioning of the software tool as a whole and the algorithms of its individual components, software implementation was carried out. The system was tested for the correctness of work in real conditions. Economic calculations of feasibility of development were carried out.

Keywords: dangerous content, information resources, system, search, analysis.

ЗМІСТ

ВСТУП

В сучасному цифровому світі, насиченому інформацією, проблема небезпечного контенту на інтернет-ресурсах стає все більш актуальною та загостреною. Швидкий розвиток технологій та широкий доступ до мережі створюють потребу в ефективних системах пошуку та аналізу небезпечного контенту.

Цифрова епоха, в яку ми ввійшли, несе із собою не тільки безмежні можливості доступу до інформації, але й виклики, пов'язані з поширенням небезпечного контенту в інтернеті [1]. Небажані вмістові елементи, такі як фейкові новини, образливий контент чи шкідливі дезінформації, стають все більшими загрозами громадській безпеці та стабільності. В цьому контексті актуальною стає проблема розробки систем, які не лише виявляють такий контент, але й забезпечують аналіз його впливу та поширення серед користувачів [2].

Задача розробки системи пошуку та аналізу небезпечного контенту інформаційних ресурсів ставить перед собою важливі завдання в контексті сучасного інтернет-простору[3]. Спроби забезпечити безпеку користувачів та визначити ступінь небезпеки інформаційного вмісту на різних платформах вимагають новаторських підходів та технологій.

Проект спрямований на створення інтегрованої системи, яка здатна виявляти різноманітний небезпечний контент, враховуючи його різні форми та вирази, а також реагувати на нові тенденції у сфері цифрової безпеки. Використання методів штучного інтелекту та машинного навчання має допомогти у покращенні точності виявлення та класифікації небезпечного контенту, а також у вивченні їхнього впливу на користувачів та суспільство.

Об’єктом дослідження є забезпечення безпеки у кіберпросторі.

Предметом дослідження є методи та засоби виявлення небезпечного контенту.

Метою даної роботи є покращення кібербезпеки шляхом створення системи для пошуку та аналізу небезпечного контенту.

Для досягнення мети потрібно розв’язати такі задачі:

– проаналізувати проблеми пошуку небезпечного контенту;

– проаналізувати методи пошуку;

– розробити метод пошуку;

– розробити архітектуру модулів пошуку та аналізу;

– розробити алгоритми роботи засобу.

Наукова новизна роботи: удосконалено метод пошуку та аналізу небезпечного контенту шляхом використання моделей обробки природньої мови, методів семантичного аналізу та аналізу контексту, що працює, в тому числі в режимі реального часу, що дозволяє оперативно виявляти та відсторонювати небезпечний контент.

Практична цінність програми полягає в такому:

- розроблено програмний засіб для пошуку та аналізу небезпечного контенту інформаційних ресурсів;
- набори сценаріїв для тестування системи;
- набори сценаріїв для використання веб-розширення користувачем для гнучкого виконання пошуку та аналізу небезпечного контенту.

Результати отримані у магістерській кваліфікаційній роботі були представлені на таких конференціях:

- Молодь у науці: дослідження, проблеми, перспективи 2024 року, Вінницького національного технічного університету [4].

1 АНАЛІЗ МЕТОДІВ ПОШУКУ ТА АНАЛІЗУ НЕБЕЗПЕЧНОГО КОНТЕНТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

1.1 Аналіз систем пошуку та аналізу небезпечного контенту

В сучасному інформаційному суспільстві проблема небезпечного контенту стоїть вкрай гостро. За Конвенцією Ради Європи про кіберзлочинність (Budapest Convention on Cybercrime), небезпечний контент включає в себе матеріали, які порушують законодавство та можуть завдати шкоди індивідуумам, групам чи суспільству в цілому. Цей контент може бути класифікований як незаконний або шкідливий[5]. Загрози, пов'язані з небезпечним контентом інформаційних ресурсів включають в себе: інтернет-шахрайство, фейкові новини та дезінформація, кібербулінг, екстремізм та тероризм, залучення до небезпечних дій [6]. Ці загрози напряму впливають на життя людини та суспільства в цілому.

Аналіз різноманітних систем включає вивчення їхніх алгоритмів, методів виявлення та класифікації небезпечного контенту, а також оцінку ефективності їхніх рішень. Порівняння функціональності, обсягу виявлення різних типів загроз, швидкості реакції на нові виклики та інших параметрів допоможе визначити те, що вже існує на ринку і які можливості є потенційно важливими для подальшого вдосконалення розроблюваної системи.

Крім того, аналіз існуючих підходів дозволить врахувати найбільш передові розробки та визначити ті напрямки дослідження, які варто акцентувати у магістерській роботі для досягнення максимальної ефективності та інноваційності розробленої системи.

Google SafeSearch є сервісом, розробленим Google для фільтрації небажаного та небезпечного контенту під час пошукових запитань [7]. Основна мета цієї системи - забезпечити безпеку користувачів, особливо дітей, під час використання пошукового двигуна Google.

Основні риси Google SafeSearch включають:

– Блокування небажаного контенту. Система фільтрує результати пошуку, вилучаючи сторінки, які містять високоякісний або порнографічний вміст, а також інші матеріали, що можуть бути непридатними для деяких аудиторій.

– Автоматична активація. Google SafeSearch може бути автоматично активованим на рівні пошукового двигуна, що дозволяє швидко та ефективно блокувати небажаний контент без необхідності додаткового втручання користувачів (рис. 1.1).

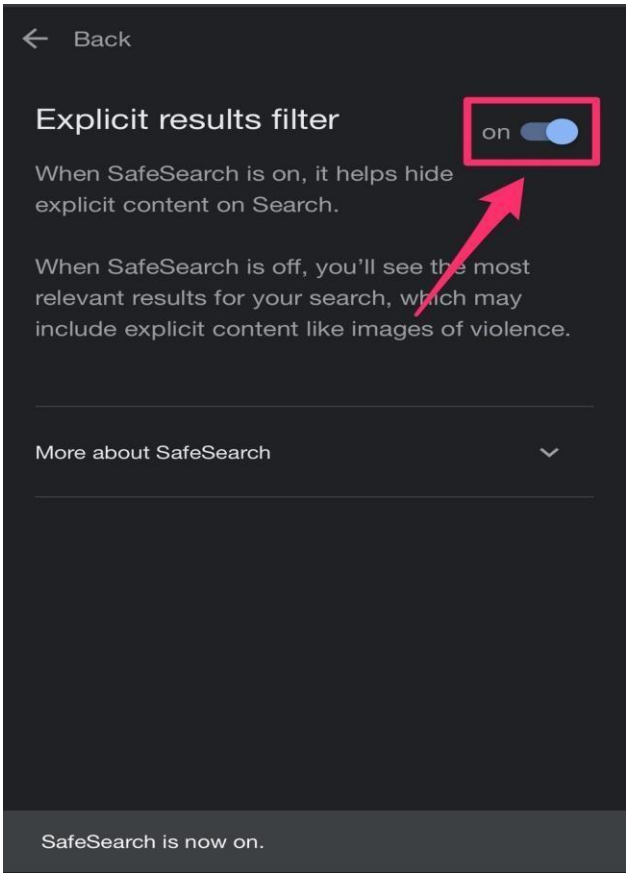


Рисунок 1.1 – Увімкнений SafeSearch на андроїд

– Можливість вручну встановлювати рівень фільтрації. Користувачі можуть вручну налаштовувати рівень фільтрації, вибираючи варіанти від "Суворий" до "Без фільтра".

– Захист від обхіду. Google SafeSearch намагається уникнути обхіду фільтрації, забезпечуючи стабільний рівень захисту.

– Використання в браузерах та мобільних пристроях. Система доступна на різних платформах, що дозволяє користувачам безпечно шукати інформацію незалежно від використовуваного пристрою.

Google SafeSearch став важливим інструментом для багатьох батьків та освітніх установ, які прагнуть забезпечити безпечне використання Інтернету дітьми.

Microsoft Defender SmartScreen - це інтегрований захист, розроблений Microsoft, який надає захист від шкідливого та небажаного вмісту при перегляді веб-сайтів [8]. Основна мета цієї системи - попередження користувачів про потенційно небезпечні сайти та запобігання загрозам кібербезпеки.

Основні характеристики Microsoft Defender SmartScreen.

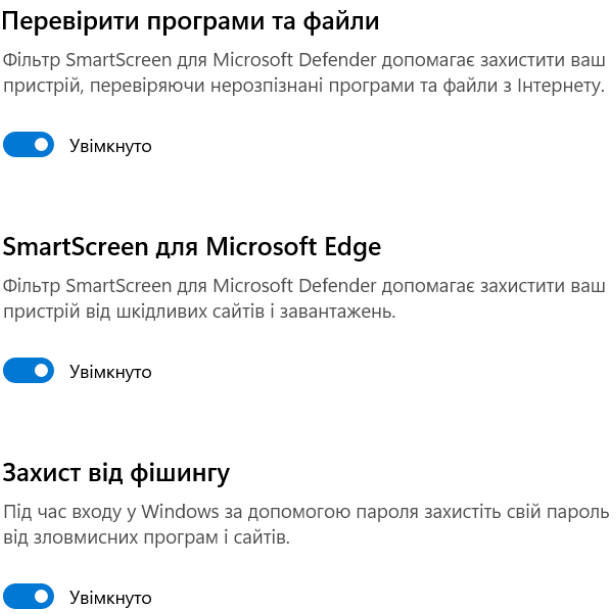


Рисунок 1.2 – Функції Microsoft Defender SmartScreen

Блокування фішингових атак. Система виявляє та блокує веб-сайти, які використовуються для фішингу, зокрема тих, що намагаються викликати введення конфіденційної інформації користувачів.

Аналіз репутації веб-сайтів. Microsoft Defender SmartScreen використовує дані про репутацію веб-сайтів для визначення, чи є вони надійними та безпечними для відвідування.

Блокування шкідливого вмісту. Система ідентифікує та блокує вміст, який може містити віруси, троянські програми та інші шкідливі елементи.

Інтеграція з браузерами Microsoft. Microsoft Defender SmartScreen вбудовано у браузери Microsoft Edge та Internet Explorer, забезпечуючи безпеку під час перегляду веб-сайтів.

Опції для підтримки корпоративних мереж. Система може бути налаштована та керуватися адміністраторами мережі для захисту корпоративних систем від кіберзагроз.

Microsoft Defender SmartScreen допомагає користувачам уникати небезпечного вмісту та залишається активним інструментом в комплексі заходів безпеки від Microsoft.

OpenDNS - це хмарний сервіс безпеки, який надає фільтрацію контенту та блокування доступу до небезпечних веб-сайтів [9]. Розроблений компанією Cisco, OpenDNS використовує технологію фільтрації DNS для захисту користувачів та мереж від шкідливого контенту та кіберзагроз.

Основні особливості OpenDNS:

Фільтрація DNS. Система використовує фільтрацію DNS для блокування доступу до сайтів, які містять небажаний або небезпечний вміст.

Категоризація веб-сайтів. OpenDNS класифікує веб-сайти за категоріями, дозволяючи адміністраторам точно налаштовувати, які типи контенту блокувати (рис. 1.3).

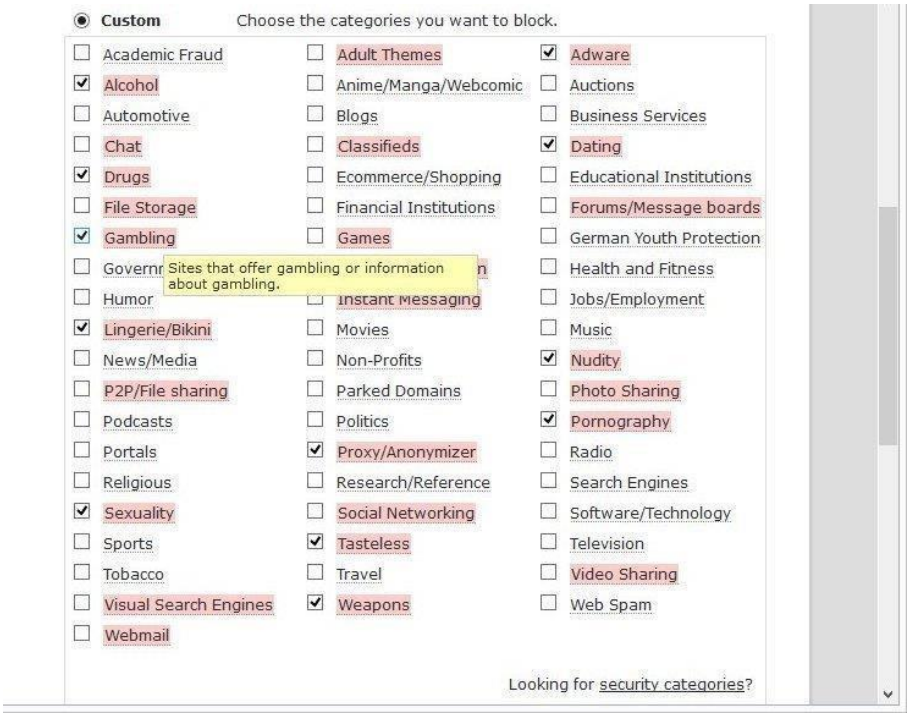


Рисунок 1.3 – Категорії контенту, що можна заблокувати

– Захист від фішингу. Система виявляє та блокує сайти, які спробують використовувати фішингові методи для отримання конфіденційної інформації.

– Статистика використання мережі. OpenDNS надає звіти та статистику щодо використання мережі, дозволяючи адміністраторам відстежувати активність та ефективність системи.

– Блокування застарілих технологій. Система може блокувати доступ до веб-сайтів, які використовують застарілі технології та потенційно вразливі до кіберзагроз.

OpenDNS є ефективним інструментом для захисту мережі від небезпечного контенту, і його використання особливо актуально для батьків, адміністраторів мереж та організацій, які прагнуть підвищити безпеку своїх користувачів.

Symantec WebFilter - це система фільтрації веб-контенту, розроблена компанією Symantec, яка надає захист від небажаного та шкідливого вмісту в Інтернеті [10]. Цей продукт призначений для підвищення безпеки користувачів та мереж, фільтруючи доступ до сайтів і контенту, які можуть становити загрозу.

Основні особливості Symantec WebFilter (рис. 1.4):

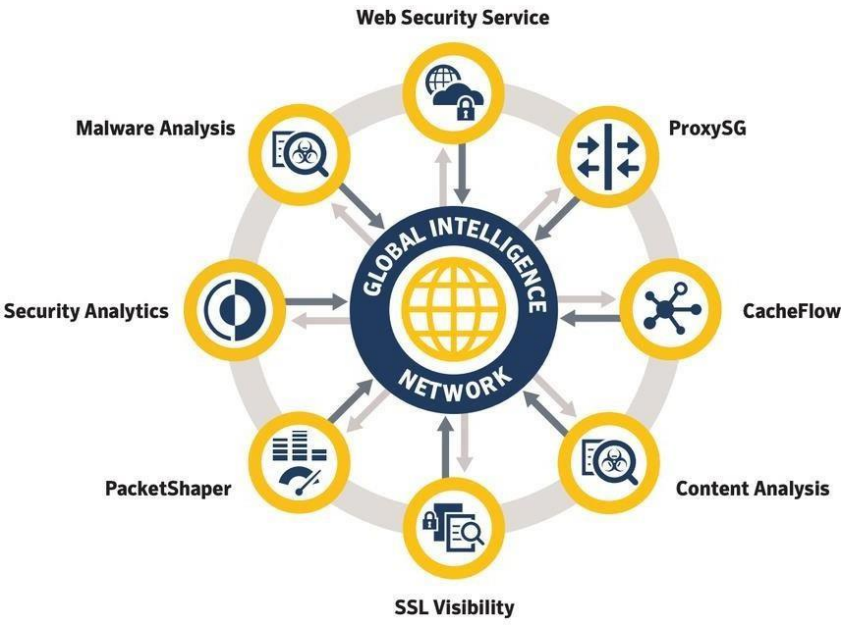


Рисунок 1.4 – Основні можливості Symantec WebFilter

Блокування небезпечного контенту. Система ідентифікує та блокує доступ до веб-сайтів, що містять шкідливий вміст, такий як віруси, троянські програми та інші загрози кібербезпеки.

Фільтрація за категоріями. WebFilter класифікує веб-сайти за різними категоріями, такими як ігри, соціальні мережі, порнографія тощо, що дозволяє адміністраторам точно налаштовувати фільтрацію в залежності від політик безпеки.

Моніторинг веб-активності. Symantec WebFilter надає інструменти для ведення журналів та моніторингу активності користувачів, що дозволяє адміністраторам стежити за використанням мережі.

Блокування анонімайзерів. Система може блокувати доступ до веб-сайтів, які намагаються приховати свою ідентичність використанням анонімайзерів.

Інтеграція з іншими рішеннями Symantec. WebFilter легко інтегрується з іншими продуктами Symantec, такими як антивірусні програми та рішення для захисту від загроз.

Symantec WebFilter є частиною розширеного пакету захисту від кіберзагроз, який допомагає організаціям та користувачам залишатися захищеними від різних веб-загроз.

Content Keeper - це інтегрована система фільтрації веб-контенту, яка надає ефективний захист від небажаного та небезпечного вмісту в Інтернеті [11]. Розроблена для корпоративного та освітнього секторів, Content Keeper пропонує ряд технологічних рішень для контролю за використанням Інтернету та забезпечення безпеки користувачів.

Основні особливості Content Keeper.

Мультирівнева фільтрація. Система використовує різноманітні методи фільтрації, включаючи ключові слова, категорії веб-сайтів та аналіз контексту, для точного виявлення та блокування небажаного вмісту.

SSL-декрипція. Content Keeper може розшифровувати трафік SSL, що дозволяє більш ефективно контролювати та фільтрувати зашифрований веб-контент.

Адаптивний контроль доступу. Система дозволяє налаштовувати правила контролю доступу в залежності від потреб організації, враховуючи конкретні політики та обмеження.

Захист від онлайн-загроз. Content Keeper виявляє та блокує шкідливий вміст, включаючи віруси, троянські програми та інші кіберзагрози.

Моніторинг активності. Система забезпечує можливість ведення журналів та аналізу активності користувачів для ефективного контролю та виявлення потенційних загроз.

Content Keeper є надійним інструментом для комплексного контролю за веб-активністю, забезпечуючи безпеку та відповідність стандартам у сфері кібербезпеки.

Web of Trust (WOT) - це розширення та веб-сервіс, які надають інформацію про безпеку веб-сайтів. Використовуючи колективний підхід, WOT дозволяє

користувачам ділитися своїми відгуками та оцінками щодо небезпечних чи ненадійних веб-ресурсів [12].

Основні характеристики Web of Trust (WOT):

Оцінка безпеки веб-сайтів. WOT використовує комбінацію автоматичного аналізу та відгуків користувачів для встановлення рівня безпеки веб-сайтів (рис. 1.6).

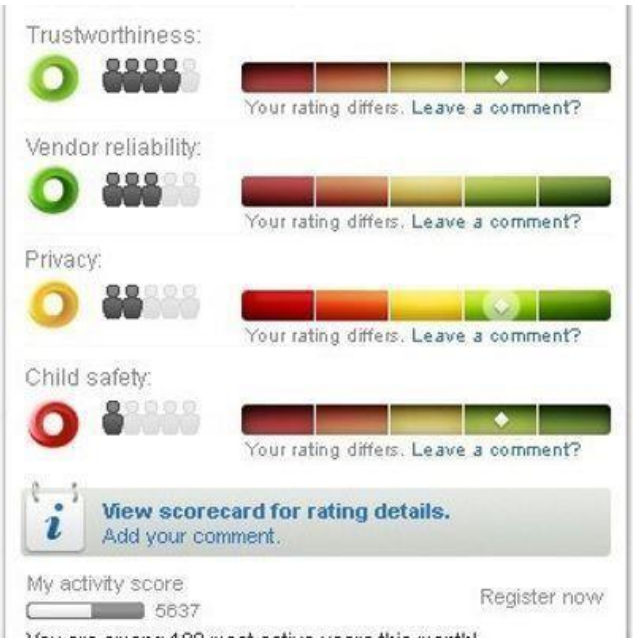


Рисунок 1.6 – Оцінка веб-сайтів

Категоризація ризиків. Сайти класифікуються за різними категоріями ризиків, такими як шахрайство, малвара, фішинг, порнографія та інші, що допомагає користувачам швидко оцінювати потенційні загрози.

Відгуки користувачів. Користувачі можуть залишати свої відгуки та коментарі щодо конкретних веб-сайтів, ділитися своєю власною експертизою та сприяти формуванню загальної думки про безпеку ресурсів.

Розширення для браузерів. WOT надає розширення для різних веб-браузерів, що дозволяє користувачам перевіряти безпеку сайтів прямо під час перегляду.

Спільнотний підхід. Взаємодія та обмін інформацією між користувачами дозволяють створити колективну базу знань про безпеку в Інтернеті.

Web of Trust є зручним інструментом для тих, хто прагне швидко оцінювати ризики, пов'язані з веб-сайтами, та довіряти думці спільноти щодо їхньої безпеки. та інших ознак.

Norton Family - це інша популярна система контролю та безпеки для дітей в онлайн-середовищі, розроблена компанією NortonLifeLock (рис. 1.7) [13].

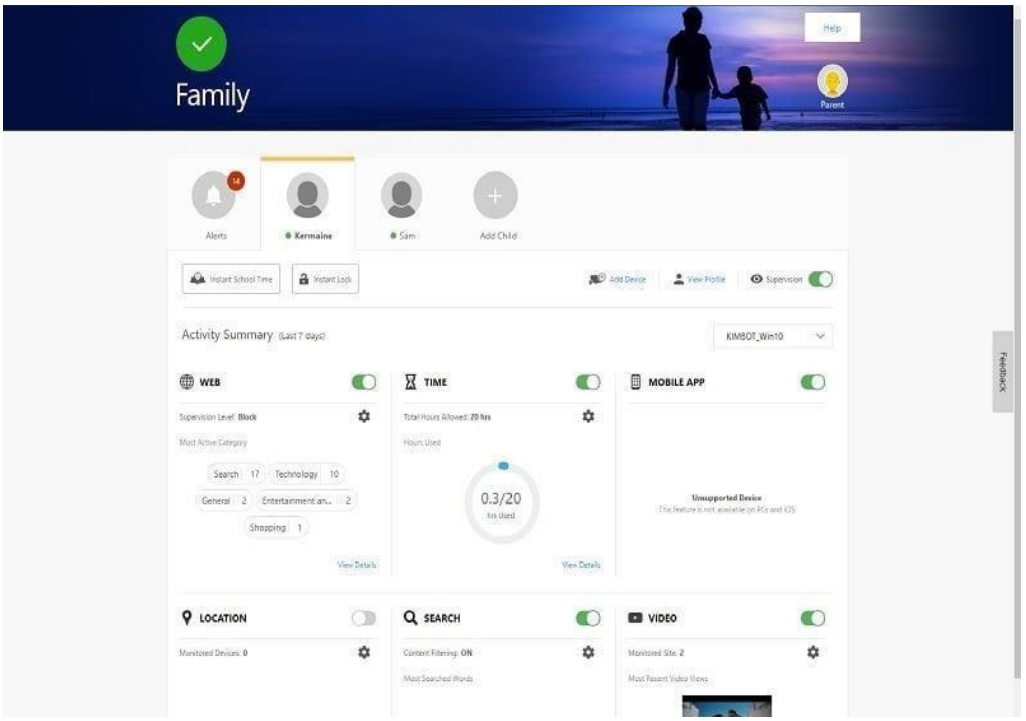


Рисунок 1.7 – Основна сторінка активного застосунку NortonFamily

Основні функції Norton Family включають.

Фільтрація контенту. Система фільтрує веб-сайти та визначає їхні категорії, щоб забезпечити безпеку під час перегляду.

Часові обмеження. Батьки можуть встановлювати часові ліміти для використання Інтернету, ігор та інших додатків.

Моніторинг активності. Norton Family веде журнал відвідувань веб-сайтів, використання програм та іншу онлайн-активність дітей.

Повідомлення про відвідування небезпечних сайтів. Батьки отримують повідомлення, якщо їхні діти намагаються відвідати сайти з обмеженим чи небезпечним вмістом.

Місцезнаходження. Система дозволяє батькам відстежувати місцезнаходження своїх дітей, щоб забезпечити їхню безпеку.

Ці інструменти є лише декількома з численних систем контролю та безпеки, розроблених для допомоги батькам в управлінні та забезпеченні безпеки онлайн-активностей своїх дітей. Це може включати перевірку фактів через сторонні джерела та контекстуальний аналіз.

Під час аналізу систем контролю та безпеки в онлайн-середовищі виявлено, що ці інструменти пропонують широкий спектр функцій для контролю контенту інформаційних рресурсів. Вони включають фільтрацію контенту, часові обмеження, моніторинг активності, SSL-декрипцію для захисту від зашифрованого трафіку, спільнотний підхід до оцінки безпеки веб-сайтів, індивідуалізацію налаштувань з урахуванням вікових особливостей, а також додаткові функції, такі як відстеження місцезнаходження.

Важливо відзначити, що кожна система має свої унікальні особливості та підходи, спрямовані на створення безпечного та відповідального онлайн-середовища. Розглядані інструменти можуть бути вибрані відповідно до конкретних потреб, надаючи можливість ефективно контролювати та захищати від різних онлайн-загроз.

У результаті аналізу різних інструментів для пошуку та аналізу небезпечного контенту інтернет ресурсів, створено таблицю з порівнянням усіх цих засобів (таб. 1.1) та визначено, що на ринку існує багато різноманітних рішень, які відповідають різним потребам суспільства. Вибір інструменту залежить від конкретних потреб та завдань.

Таблиця 1.1 – Порівняння інструментів для виявлення небезпечного контенту

Характеристика	NortonFamily	WOT	ContentKeeper
Фільтрація контенту	Так	Так	Так
Часові обмеження	Так	Ні	Так
Моніторинг активності	Так		
SSL-декрипція	Ні	Ні	Так
Спільнотний підхід	Ні	Так	Ні
Індивідуалізація налаштувань	Так	Ні	Так
Захист від онлайн-загроз	Так	Ні	Так
Категоризація ризиків	Так	Так	НІ
Мультирівнева фільтрація	Ні	Ні	Так

1.2 Аналіз методів пошуку та аналізу небезпечного контенту

Аналіз методів пошуку та аналізу небезпечного контенту є важливою частиною розробки систем, спрямованих на забезпечення безпеки користувачів у віртуальному середовищі. Різні методи використовуються для ефективного виявлення та фільтрації небезпечного контенту, щоб запобігти його негативним наслідкам.

Метод аналізу ключових слів та фраз виявляє небезпечний контент на основі спеціально визначених та попередньо встановлених ключових слів або фраз, які пов'язані з певним видом загроз або небажаного вмісту [14]. Цей підхід використовує статичні фільтри, які перевіряють вміст на відповідність цим ключовим словам або фразам без врахування контексту чи семантики.

Програми, що використовують цей метод, можуть сканувати тексти, веб-сторінки, повідомлення чи інші види інформації на наявність конкретних слів або фраз, які можуть свідчити про шкідливий чи небажаний характер контенту. Такий аналіз може включати в себе не тільки експліцитні ключові слова, але і їхні синоніми чи варіації, щоб забезпечити більш широкий охоплення можливих варіантів небезпечного вмісту.

Цей метод може бути ефективним для виявлення конкретних форм небезпеки або небажаного вмісту, але водночас він може бути обмеженим в здатності виявляти нові форми загроз або ті, які використовують нові вирази. Важливо враховувати контекст та змістовні особливості, оскільки без цього може виникнути надмірна чутливість або недостатня точність в роботі системи.

Основні характеристики цього методу включають

- Системи використовують ключові слова та фрази, щоб ідентифікувати можливі небезпечні або небажані вміст.
- Простий та швидкий спосіб виявлення. Допомагає виявляти вміст за конкретними критеріями.
- Може бути неефективним виявлення нових або змінених формулювань небезпечного контенту.

Метод машинного навчання та аналізу поведінки є передовим підходом до виявлення небезпечного контенту. В основі цього методу лежить використання алгоритмів машинного навчання, які навчаються розпізнавати та класифікувати різні форми небезпеки на основі великого обсягу даних. Системи, які використовують цей підхід, навчаються на великій кількості вхідних даних, що можуть включати текстову інформацію, зображення, відео та інші типи контенту. Вони аналізують ці дані, вивчаючи патерни та характеристики, які вказують на небезпечний вміст. Доцільно враховувати не лише сам вміст, але і поведінку користувачів, так як це дозволяє виявляти аномалії та несподівані зміни.

Важливою особливістю цього методу є його здатність адаптуватися до нових форм загроз і швидко реагувати на зміни в інтернет-середовищі. Використання машинного навчання дозволяє створювати моделі, які можуть підтримуватися та оновлюватися з часом, щоб враховувати нові тренди та загрози.

Необхідно враховувати, що для ефективності цього методу потрібно значну кількість навчальних даних, і важливо регулярно оновлювати моделі для збереження їхньої актуальності в змінному інтернет-середовищі.

Методи машинного навчання для виявлення пошуку небезпечного контенту є досить ефективними і можуть використовуватися для автоматизованого виявлення

недостовірної інформації в реальному часі. Однак він вимагає наявності великої кількості навчальних даних та впровадження складних алгоритмів машинного навчання.

Метод аналізу URL-адрес та доменів базується на перевірці ідентифікаторів веб-ресурсів, а саме URL-адресів та доменних імен, на належність відомим спискам шкідливих ресурсів. Використання цього методу сприймається як ефективний спосіб блокування доступу до відомих джерел небезпечного контенту.

Програми, що використовують цей метод, можуть аналізувати URL-адреси, які користувачі відвідують, або з якими вони спробують взаємодіяти, та порівнювати їх із заздалегідь складеними чорними списками доменів або конкретних адрес шкідливих сайтів. Це дозволяє системам оперативно виявляти та блокувати доступ до вже відомих джерел небезпеки.

Однак важливо враховувати, що цей метод може бути менш ефективним у виявленні нових або персоналізованих загроз, оскільки він зосереджений на відомих векторах атак та шкідливих ресурсах. Також відзначається можливість блокування доступу до безпечних ресурсів, якщо вони помилково визначаються як потенційно небезпечні за ключовими критеріями.

Основні властивості цього методу включають:

- блокування за відомими джерелами. Метод використовує чорні списки доменів та URL-адрес для виявлення та блокування доступу до відомих шкідливих джерел;
- швидкість виявлення. Здатний оперативно реагувати на відомі загрози через використання заздалегідь складених списків;
- очікувані обмеження. Може бути менш ефективним у виявленні нових, не відомих загроз, оскільки аналізує тільки відомі чи частково відомі джерела.
- ризик блокування безпечних ресурсів. Існує ризик блокування доступу до безпечних ресурсів, якщо вони помилково асоціюються з шкідливими доменами або URL-адресами;
- обмежений контекст. Оцінює загрози на основі самого ідентифікатора, не беручи до уваги контекст або семантику вмісту;

– стійкість до змін. Стійкий до змін у виявленні шкідливих джерел, якщо немає нових вхідних даних чи оновлень списків.

Метод аналізу метаданих мультимедійних файлів базується на тому, щоб виявляти небезпечний контент шляхом розглядання додаткової інформації, що приєднана до зображень, відео чи інших мультимедійних файлів. Під час цього аналізу враховуються різноманітні метадані, такі як геолокація, час створення, авторська інформація та інші деталі, які можуть вказувати на особливості контенту та його потенційну небезпеку.

Важливим є те, що цей метод не обмежується лише аналізом самих вмістових елементів, але використовує контекстуальні відомості, щоб зрозуміти можливі наслідки взаємодії з такими мультимедійними файлами. Наприклад, він може ідентифікувати конфіденційні дані, приховані в метаданих, або виявляти небажані елементи, які можуть бути приховані на рівні додаткової інформації.

Основні властивості методу аналізу метаданих мультимедійних файлів:

– контекстуальний аналіз. Поєднання аналізу метаданих та самого вмісту для повного розуміння можливих загроз;
– виявлення конфіденційних даних. Здатність ідентифікувати конфіденційні або небажані елементи, приховані в метаданих;
– комплексний підхід. Врахування різних аспектів інформації, що допомагає виявляти небезпечний контент та негативні наслідки його використання.

Метод аналізу структури тексту та семантики використовується для виявлення небезпечного контенту, аналізуючи не лише окремі слова чи фрази, але й їхні взаємозв'язки та семантичний зміст. Цей підхід орієнтований на розуміння суті текстового вмісту та виявлення можливих загроз на основі його структури.

Під час використання цього методу система аналізує, як слова та фрази взаємодіють одне з одним, визначає контекст та спробує розглядати текстовий вміст як цілісну структуру. Це дозволяє виявляти небезпечний контент, який може бути виражений не тільки конкретними ключовими словами, але й загальним сенсом чи намірами.

Основні властивості методу аналізу структури тексту та семантики:

- синтаксичний та семантичний аналіз. Врахування синтаксичної структури речень та їхньої семантики для виявлення можливих загроз.
- контекстуальне розуміння. Оцінка тексту у контексті для забезпечення більш точного виявлення загроз та небажаного вмісту.
- аналіз взаємозв'язків. Виявлення зв'язків між словами та фразами, що може вказувати на наявність небезпеки чи шкідливого змісту.
- адаптивність до контексту. Здатність адаптуватися до змінного семантичного контексту мови та виявляти загрози, навіть якщо вони виражені в нових формулюваннях чи виразах.

Метод аналізу поведінки та взаємодії користувачів здійснюється за допомогою алгоритмів машинного навчання та аналізу патернів, щоб виявляти небезпечний контент на основі змін у поведінці користувачів та їхніх взаємодій з інформаційним середовищем.

Цей метод враховує не тільки статичний аналіз контенту, а й динамічні аспекти, пов'язані з активністю користувачів. Шляхом вивчення та аналізу, як користувачі спілкуються з інформаційним вмістом, система може виявляти зміни у звичайному поведінковому патерні, які можуть свідчити про наявність небезпеки.

Основні властивості методу аналізу поведінки та взаємодії користувачів:

- динамічний аналіз. Аналіз змін у поведінці користувачів в реальному часі.
- вивчення патернів. Використання алгоритмів машинного навчання для виявлення незвичайних або відмінних патернів в поведінці.
- залежність від контексту. Врахування контекстуальних факторів, таких як час доби, локація користувача та інші, для точного виявлення загроз.
- виявлення аномалій. Здатність виявляти аномальні зміни у поведінці, що можуть бути індикаторами небезпечного вмісту.
- адаптивність. Змога адаптуватися до нових форм небезпечного контенту та змін у патернах користувачів.

Цей підхід дозволяє створювати системи, які не тільки реагують на відомі загрози, але й мають здатність вчиться та адаптуватися до нових векторів атак.

Аналіз цих методів дозволяє врахувати різні аспекти та підходи до виявлення небезпечного контенту, що може бути важливим для створення комплексних систем безпеки в інтернеті.

Аналіз методів пошуку та аналізу небезпечного контенту демонструє широкий перелік доступних методів для вирішення проблеми з пошуком небезпечного контенту в інформаційних ресурсах.

1.3 Постановка задачі

Розробка нового засобу для пошуку та аналізу небезпечного контенту інформаційних ресурсів у вигляді веб-додатку є дуже актуальною для забезпеченя суспільства від кіберзагроз. Мета проекту полягає в створенні інтерактивної платформи, де користувачі зможуть перевіряти бажані сайти на наявність небезпечного контенту, буде маркування вже перевірених веб-сайтів.

Серед основних завдань цього проекту включається розробка інтерфейсу, можливість аналізу інформації, інтеграція спільноти, створення механізмів для маркування небезпечного контенту.

Після детального аналізу різноманітних систем пошуку та аналізу небезпечного контенту, а також розгляду різних методів виявлення, враження від формування загального уявлення виявилося насиченим та динамічним.

Різні методи, такі як аналіз структури тексту, метаданих, поведінки користувачів і машинного навчання, представляють собою ефективний підхід до розв'язання проблеми виявлення небезпечного контенту. Кожен з них має свої переваги та обмеження, але в їх комбінації може бути знайдено баланс для створення більш надійних та адаптивних систем.

У цілому, зазначений аналіз дозволяє визначити важливі напрями для подальших досліджень та розробки в галузі систем пошуку та аналізу небезпечного контенту, зокрема, розробки більш ефективних та адаптивних підходів для захисту інтернет-середовища.

2 ВИБІР МЕТОДУ ПОШУКУ ТА АНАЛІЗУ НЕБЕЗПЕЧНОГО КОНТЕНТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

2.1 Обгрунтування вибору методів аналізу тексту та семантики

В цьому підрозділі досліджуються методи що дозволяють розуміти смислове наповнення текстового вмісту та визначити їхню застосовність у контексті виявлення небезпечного вмісту. Розгляд цих методів сприятиме глибшому розумінню їхньої ефективності та визначенню переваг та обмежень у контексті комплексної системи аналізу безпеки інформаційних ресурсів.

В рамках аналізу методів аналізу тексту та семантики розглядаються техніки, спрямовані на глибоке розуміння смислового змісту текстового матеріалу. Основна мета полягає в розрізненні між різними відтінками значень, виявленні виразних та амбігвітних конструкцій, що можуть вказувати на наявність потенційно небезпечного контенту.

Методи аналізу тексту дозволяють враховувати не лише окремі слова чи фрази, але й їхні взаємозв'язки, контекст та семантичний зміст. Це важливо для розуміння загального сенсу висловлювань та виявлення виразів, які можуть мати двозначний чи прихований характер.

Аналіз семантики включає в себе розгляд того, як слова та фрази взаємодіють між собою та як їхнє спільне використання може впливати на смислове навантаження тексту[15]. Цей підхід дозволяє виявляти не лише явні ознаки небезпечного контенту, але і підтексти, які можуть бути виявлені лише при уважному аналізі контекстуальних зв'язків.

Загальний аналіз методів аналізу тексту та семантики покликаний визначити їхню ефективність у виявленні потенційно небезпечного контенту та визначити їхні переваги та обмеження в контексті системи пошуку та аналізу інформаційних ресурсів.

Метод аналізу метаданих спрямований на вивчення та інтерпретацію додаткової інформації, що супроводжує мультимедійний контент, такої як

фотографії чи відео. Зокрема, досліджується, як аналіз метаданих може допомогти виявити потенційно небезпечний контент та класифікувати його залежно від різноманітних параметрів.

Цей підхід включає розгляд інформації, що супроводжує файл, такої як дата та місце зйомки, параметри камери чи спосіб обробки зображення. Аналіз метаданих важливий для виявлення можливих аномалій, які можуть свідчити про фальсифікацію або несанкціоноване використання контенту.

Важливо врахувати, що аналіз метаданих може надати системі додатковий контекст та дозволити визначити відповідність мультимедійного вмісту конкретним критеріям безпеки. Однак в той же час, методи аналізу метаданих можуть бути обмежені в разі відсутності або недостатньої достовірності метаданих.

Такий підхід до аналізу метаданих спрямований на виявлення можливих загроз у мультимедійному вмісті та визначення їхнього характеру для подальшого класифікування та відповідного реагування системи.

Розглядаючи методи аналізу структури тексту та семантики, визначаємо їхні ключові аспекти. Основний метод базується на розумінні та інтерпретації мовленнєвих конструкцій у текстових матеріалах. Це включає в себе розпізнавання синтаксичних структур, визначення значень слів та виявлення контекстуальних зв'язків. Особлива увага приділяється розумінню імпліцитних сенсів та аналізу емоційного забарвлення висловлювань.

У випадку аналізу метаданих мультимедійних файлів, метод включає вивчення інформації, що супроводжує фотографії чи відео [16]. Зазвичай це включає дату та місце зйомки, параметри камери, а також дані про обробку зображень. Системи використовують цю інформацію для виявлення аномалій, порушень цілісності зображень та виявлення можливих змін у мультимедійному контенті.

Обидва методи базуються на алгоритмах машинного навчання та обробці природної мови для ефективного аналізу великих обсягів даних. Аналітичні засоби використовуються для автоматичного визначення смислових аспектів тексту чи метаданих та класифікації їх за заданими критеріями безпеки.

Важливим елементом обох методів є поєднання аналітичної потужності та контекстуального розуміння для досягнення високого рівня точності та надійності у виявленні небезпечного контенту на інформаційних ресурсах.

Під час вибору оптимальних методів необхідно врахувати кілька ключових аспектів (рис. 2.1).

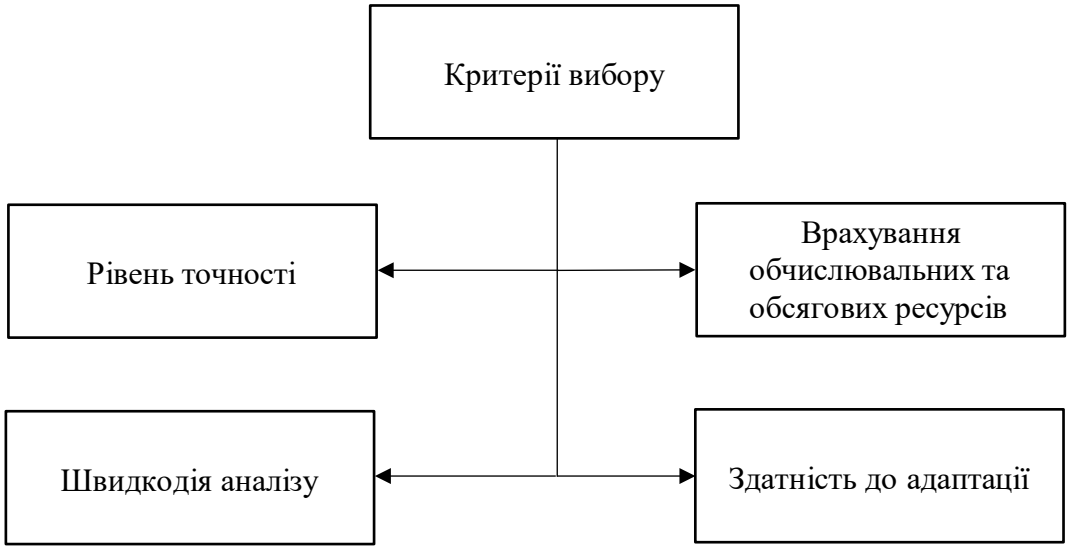


Рисунок 2.1- Схема критеріїв вибору оптимальних методів

Рівень точності методів виявлення небезпечного контенту є критичним параметром. Система має визначити, які методи забезпечують найвищий рівень точності в розпізнаванні та класифікації потенційно небезпечного контенту.

Врахування обчислювальних та обсягових ресурсів є важливим аспектом. Система має оцінити, наскільки витратні є обрані методи, і чи вони можуть бути ефективно використані в конкретних умовах.

Швидкодія аналізу важлива для реального часу реакції на потенційні загрози. Обрані методи повинні бути ефективними з точки зору обробки інформації в реальному часі.

Здатність методів адаптуватися до різних типів контенту та еволюції методів шахрайства є важливою. Система повинна забезпечити гнучкість та актуальність у змінюваних умовах.

Важливо розглянути, як обрані методи можуть взаємодіяти між собою для створення синергії та підвищення загальної ефективності системи.

Аналіз цих аспектів дозволяє системі обрати найбільш оптимальний та ефективний підхід до аналізу структури та семантики для виявлення небезпечного контенту на інформаційних ресурсах.

2.2 Комбінація методів для комплексного підходу

Важливість об'єднання різних методів для створення комплексного підходу до системи пошуку та аналізу небезпечного контенту, є досить суттєвою. Ця комбінація методів забезпечує високий рівень ефективності та дозволяє системі краще впоратися з різними викликами.

Переваги такого комплексного підходу виявляються у здатності кожного методу доповнювати і підтримувати інші. Аналіз структури та семантики тексту, разом із вивченням метаданих мультимедійних файлів, створює повніший погляд на контент і робить систему більш чутливою та надійною.

Важливим аспектом є не лише використання різних методів, а й їхня взаємодія. Результати одного методу можуть служити вхідними даними для іншого, що сприяє комплексному розумінню контенту. Адаптабельність до змін та здатність системи виявляти аномалії стають ключовими факторами для довгострокового успіху.

Цей комплексний підхід не тільки дозволяє ефективно виявляти небезпечний контент, а й надає можливість системі постійно вдосконалюватися. Аналіз результатів сприяє постійному вдосконаленню методів та адаптації до нових викликів у сфері безпеки інформації.

Для прикладу комплексного підходу до системи пошуку та аналізу небезпечного контенту на інформаційних ресурсах обрано систему в якій використовуються три основні методи: аналіз структури тексту, аналіз семантики та аналіз метаданих мультимедійних файлів.

Аналіз структури тексту. Цей метод використовує алгоритми обробки природної мови для виявлення синтаксичних та структурних елементів текстового контенту. Визначаються ключові слова, синтаксичні конструкції та зв'язки між словами.

Аналіз семантики. Метод аналізу семантики глибоко розуміє смисловий контекст текстового матеріалу. Використовуються алгоритми машинного навчання для розрізнення смислів слів, класифікації висловлювань за їхньою інтенцією та виявлення емоційного забарвлення тексту.

Аналіз метаданих мультимедійних файлів. Цей метод сконцентрований на інформації, що супроводжує фотографії чи відео. Вивчається дата та місце зйомки, параметри камери, а також дані про обробку зображень. Виявлення аномалій у цій інформації може свідчити про можливий небезпечний вміст.

Взаємодія методів. Результати аналізу структури тексту можуть служити вхідними даними для аналізу семантики, а відомості з метаданих мультимедійних файлів можуть доповнювати інформацію, отриману від аналізу семантики та структури тексту. Така взаємодія покращує повноту та достовірність аналізу.

Цей комплексний підхід надає системі великий потенціал для виявлення різноманітних загроз та забезпечення більш високого рівня безпеки на інформаційних ресурсах.

2.3 Аспекти безпеки та конфеденційності

Аспекти безпеки та конфіденційності в контексті системи пошуку та аналізу небезпечного контенту на інформаційних ресурсах є невід'ємною частиною її проектування та функціонування. Враховуючи чутливість інформації та потенційні загрози безпеки, система повинна дотримуватися ряду ключових принципів:

– Конфіденційність інформації.

Забезпечення конфіденційності даних в системі є надзвичайно важливим аспектом. Всі дані, що обробляються та зберігаються системою, повинні бути надійно захищені від несанкціонованого доступу. Використання сучасних методів

шифрування та контролю доступу дозволяє уникати витоку конфіденційної інформації.

– Захист від кібератак.

З урахуванням того, що система операцій у віртуальному просторі, важливо мати міцний захист від кібератак. Це включає в себе використання міжмережевих екранів систем виявлення вторгнень, антивірусного захисту та інших технологій для забезпечення безпеки під час обміну даними та взаємодії з іншими системами.

– Ідентифікація та автентифікація. Засоби ідентифікації та автентифікації грають ключову роль у забезпеченні безпеки. Забезпечення доступу лише авторизованим користувачам та системам є фундаментальною задачею. Використання паролів, біометричних методів та двофакторної аутентифікації є поширеними практиками для захисту від несанкціонованого доступу.

– Аудит та моніторинг. Система повинна вести детальний аудит та моніторити активність для виявлення будь-яких підозрілих або нестандартних подій. Це дозволяє оперативно реагувати на потенційні загрози та забезпечує прозорість у використанні системи.

– Безпека мережі та зв'язку. Захист інформації під час передачі через мережу є важливим. Використання протоколів шифрування, VPN-з'єднань та інших заходів дозволяє уникати перехоплення та зміни даних під час їхнього переміщення через мережу.

– Резервне копіювання та відновлення. Забезпечення можливості резервного копіювання та відновлення даних в разі втрати або пошкодження є критичним. Це гарантує можливість відновлення системи та інформації в найкоротший термін після можливого інциденту.

Загальний підхід до безпеки та конфіденційності включає в себе комплекс заходів, що враховують технічні, організаційні та правові аспекти, щоб забезпечити високий рівень захисту інформації в системі.

2.4 Висновки з розділу

У даному розділі було розглянуто різні методи та підходи, що використовуються для пошу та аналізу небезпечого контенту інформаційних ресурсів, а також було відмічено важливу роль комплексного підходу до пбудови системи. В умовах сучасного віртуального середовища, де буде працювати система, важливо систематично вдосконалювати заходи захисту від кібератак.

Методи пошуку та аналізу небезпечного контенту розглядались в контексті аналізу структури тексту та семантики, взаємодії користувачів та методів машинного навчання. Важливим є розуміння, що жоден з цих методів не є універсальним, а отже необхідно використовувати комплексний підхід.

Загальною тенденцією є поєднання різних методів та підходів для досягнення найкращих результатів. Важливо також враховувати постійну еволюцію стратегій поширення небезпечного контенту, що вимагає постійного вдосконалення систем виявлення.

3 РОЗРОБКА СТРУКТУРНОЇ СХЕМИ СИСТЕМИ ПОШУКУ ТА АНАЛІЗУ НЕБЕЗПЕЧНОГО КОНТЕНТУ

3.1 Узагальнена архітектура

Система призначена для пошуку та аналізу небезпечного контенту інформаційних ресурсів. Дана система містить такі блоки:

- модуль взаємодії з користувачем, у вигляді веб-розширення;
- модуль збору даних;
- модуль фільтрації;
- модуль аналізу тексту та зображень;
- модуль класифікації;
- модуль звітування;
- база даних для зберігання проаналізованого контенту.

Узагальнена архітектура системи представлена нижче (рис. 3.1).

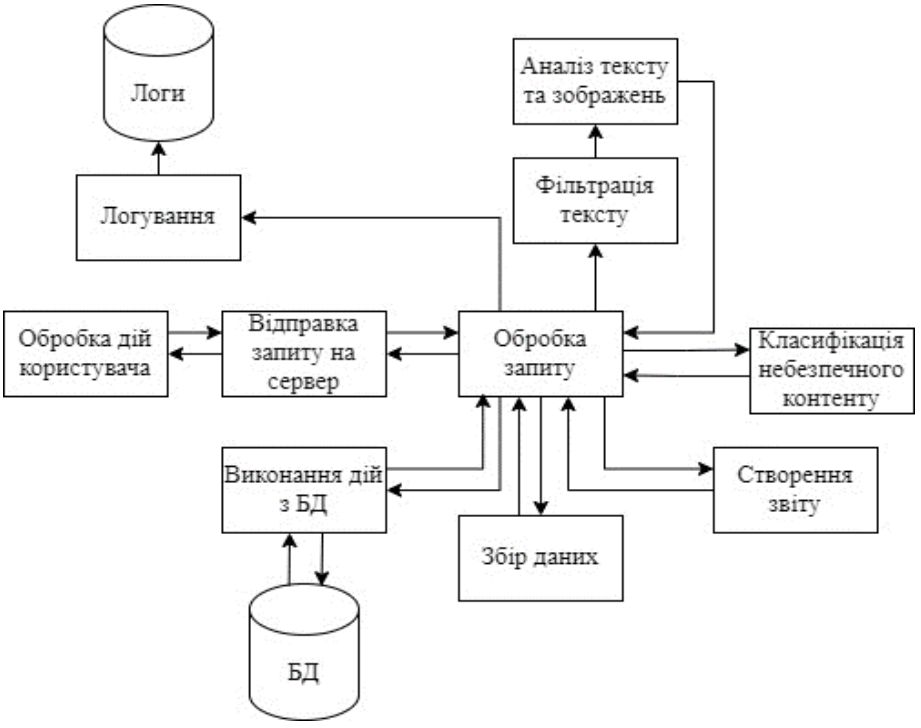


Рисунок 3.1 – Узагальнена архітектура системи

Потрібно розробити інтерфейс на основі веб-додатку. Для зручності

взаємодії додати навігаційну панель та довідку з користування.

Для збору інформації буде використовуватися веб-скрапінг через бібліотеки, такі як Scrapy[17]. Для роботи з API можна використати бібліотеку requests[18]. Важливо також врахувати питання безпеки і легальності збору інформації. У деяких випадках збір даних може супроводжуватися обмеженнями, і важливо дотримуватися правових норм і політик використання даних.

У модулі фільтрації відбуватиметься відсіювання контенту, що не є небезпечним. Це може бути базовий фільтр на основі ключових слів або більш складні алгоритми фільтрації[19].

Модуль аналізу тексту та зображень використовуватиме методи машинного навчання для аналізу текстового та візуального контенту, виявлення відмінностей, специфічних слів, образів чи контексту, які можуть вказувати на небезпеку[20].

Модуль класифікації визначатиме наскільки контент небезпечний і відповідно до цього встановлюватиме рівень загрози. З цією метою можна ввести градацію небезпечності контенту та розділяти його.

Модуль звітування та дії відповідатиме за створення звітів.

3.2 Модуль взаємодії з користувачем

Користувацький інтерфейс розробляється у вигляді веб-додатку. Вибір такого рішення пов’язаний з тим, що це є дуже зручно і ефективно. Крім того, за допомогою цього рішення вирішується питання кросплатформенності для цього засобу, так як може запускатися у браузерах на різних платформах. Для зручності користувачів потрібно додати навігаційну панель.

Дизайн пропонується розробити простим та інтуїтивно зрозумілим. Для реалізації серверної частини обрано серверну мову програмування node.js[20].

Блок-схема роботи модуля представлена на рис. 3.2.

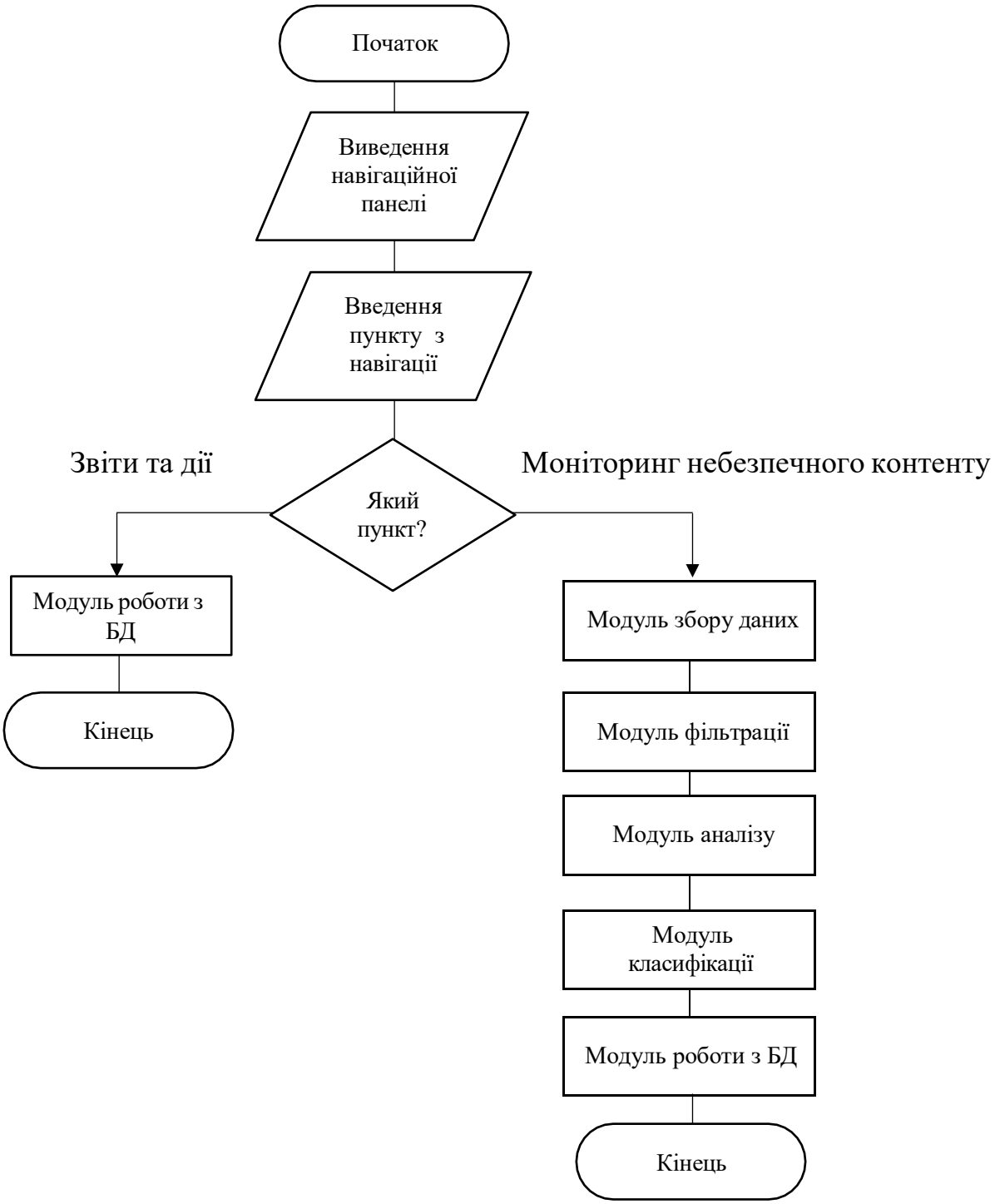


Рисунок 3.2 – Схема роботи модулю взаємодії з користувачем

Цей модуль пов’язує між собою інші модулі застосунку для повноцінноїроботи.

3.3 Модуль збору даних

Модуль збору даних - це компонент системи, який відповідає за отримання інформації з різних джерел. Його основна робота полягає у відвідуванні або отриманні доступу до визначених джерел і збиранні необхідної інформації.

Схема роботи модулю збору даних зображена на рисунку 3.3.

Рисунок 3.3 – Схема роботи модуля збору даних

Першим етапом є визначення ресурсів чи джерел, з яких потрібно отримати дані. Це може бути веб-сайт, API, база даних, файли тощо.

Далі модуль встановлює з’єднання з джерелами. Реалізація механізму для

взаємодії з джерелами включає веб-скрапінг (вилучення даних з веб-сторінок), виклики до API, підключення до бази даних чи інші методи збору інформації.

Наступним кроком буде обробка та фільтрація даних - очищення, форматування та відфільтрування зібраної інформації для подальшого використання. Це може включати видалення непотрібних елементів, перетворення даних у потрібний формат.

Передача попередньо обробленої, зібраної інформації відбуватиметься у модуль фільтрації.

3.4 Модуль роботи з базами даних

Для реалізації всіх функції засобу потрібно мати доступ до баз даних, для вирішення даної задачі обрано MySQL. Через переваги у простоті роботи та швидкості підключення і запису в базу даних.

Для зберігання та вивчення знайденого небезпечного контенту інормаційних ресурсів пропонується додати до системи модуль роботи з базами даних.

Для реалізації цієї задумки потрібно написати модуль запису результатів перевірки в базу даних, додати токен бази даних в конфігураційний файл, а також відкрити доступ до IP-адреси в налаштуваннях самої бази даних.

Після перевірки модуль отримуватиме веб-ресурс, що перевірявся, та результат пошуку небезпечного контенту в ньому.

Загальна схема роботи модулю запису в базу даних представлена на рисунку 3.4.

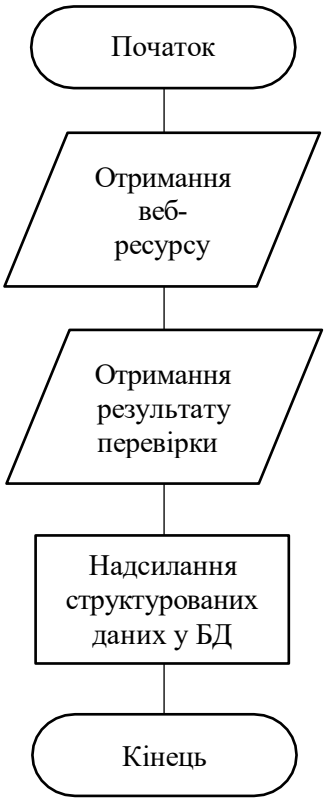


Рисунок 3.4 – Схема роботи модулю запису в базу даних

Робота з базою даних допоможе забезпечити запис виявленого небезпечного контенту, результати перевірок, а також дасть змогу звернутися до даних перевірки після повторного аналізу.

3.5 Модуль фільтрації

Модуль фільтрації відповідає за обробку отриманих даних та відсіювання чи класифікацію інформації згідно з певними критеріями чи правилами. Основна його мета - відокремлення та видалення звичайної, безпечної інформації, для пошуку небезпечного контенту.

Архітектура модулю спрямована на поєднання пошуку за ключовими слова і фразами та використання методів обробки природної мови, для аналізу тексту та його емоційної тонованості.

Першим етапом фільтрації є створення правил, які визначають який тип інформації вважається небажаним, неприйнятним або небезпечним.

Після отримання даних модуль аналізує їх з використанням заданих критеріїв. Це може включати пошук специфічних фраз, перевірку наявності певних елементів чи порівняння зі списками заборонених чи небажаних образів чи слів.

На основі зазначених правил модуль фільтрації відокремлює небажаний або небезпечний контент від прийнятного. Це може бути виконано через маркування даних, відфільтрування або класифікацію на основі рівня ризику.

Після фільтрації модуль видаляє прийнятний контент, а відфільтрований небезпечний відправляється в модуль аналізу тексту та зображень.

Схема роботи модулю фільтрації зображена на рисунку 3.5.

Рисунок 3.5 – Схема роботи модулю аналізу

Даний модуль розробляється з метою зменшення навантаження на модуль аналізу та системи в цілому. Це досягається за рахунок зменшення обсягу контенту, який буде проаналізовано під час роботи модуля аналізу тексту та зображень.

3.6 Модуль аналізу тексту та зображень

Модуль аналізу тексту і зображень для пошуку небезпечного контенту використовує різні алгоритми та методи для виявлення небажаного або небезпечного матеріалу в текстових або візуальних даних.

Текст розбивається на окремі слова чи токени для подальшої обробки. При цьому застосовуються методи очищення від спецсимволів, перетворення на нижній регістр і видалення стоп-слів.

Моделі обробки природної мови (NLP) використовуються для виявлення емоційного тону, класифікації тексту за тематикою та виявлення певних ключових слів чи фраз, які можуть вказувати на небезпечний контент (наприклад, насильство, розпалювання ненависті тощо).

Порівняння тексту з певними наборами ключових слів або шаблонами, які вказують на небезпечний вміст. Це може включати образливі слова, фрази, специфічні лексичні конструкції та інше.

Аналіз метаданих зображення (наприклад, EXIF-даних) та контексту, який оточує зображення, може допомогти виявити небажаний контент.

Ці методи будуть використовуватися в комбінації для створення ефективного модулю аналізу тексту та зображень для пошуку небезпечного контенту. Точність і ефективність аналізу зазвичай залежать від якості обраних моделей та правильного підбору параметрів та критеріїв фільтрації.

Схема роботи модулю аналізу тексту та зображень представлена на рисунку 3.6.

Рисунок 3.6 – Схема роботи модулю аналізу тексту та зображень

Після детального аналізу результати, а також визначений тип небезпечного контенту відправляються в модуль класифікації.

3.7 Модуль класифікації

Для зручності використання системи пошуку та аналізу небезпечного контенту прийнято рішення додатково розробити модуль класифікації небезпечного контенту.

Основна задача роботи модулю полягає в встановленні оцінки знайденого небезпечного контенту на веб-ресурсі. Це зроблено для зручності формування звіту та можливості в подальшому формувати графіки знайденого того чи іншого виду небезпечного контенту.

Для класифікації небезпечного контенту пропонується використовувати наступну градацію (табл. 3.1).

Таблиця 3.1 – Градація для класифікації небезпечного контенту

Рівень	Назва	Опис
0	Безпечний	Рівень призначений для ресурсів, які абсолютно безпечні та не містять жодного неприйнятного чи небезпечного контенту. Такий рівень дозволяє користувачам впевнено переглядати вміст без ризику зіткнення з будь-якими формами шкідливого впливу.
1-3	Відносно безпечний	Рівні враховують можливість наявності неприйнятного контенту, такого як образи чи нецензурна лексика, яка не є безпосередньо небезпечною, але може викликати дискомфорт чи обурення деяких користувачів. Визначення таких рівнів дозволяє виділити ресурси, які взагалі безпечні, але містять обмежений образливий вміст.
4-7	Потенційно небезпечний	На цих рівнях розташовані ресурси, які можуть містити контент, що розпалює ворожнечу, екстремізм чи заклики до насильства. Цей вміст може впливати на думки і переконання користувачів, спонукати їх до негативних дій, тому важливо визначати його наявність та попереджувати можливі негативні наслідки.
8-10	Небезпечний	На цих рівнях розташовані ресурси з високо ризиковим вмістом, таким як порнографія, кадри вбивств чи знущань, поширення наркотичних речовин. Цей контент може призводити до серйозних фізичних чи психологічних наслідків для користувачів, тому обмеження доступу до таких ресурсів є важливим для збереження безпеки та здоров'я користувачів.

Схема роботи модулю класифікації зображена на рисунку 3.7.

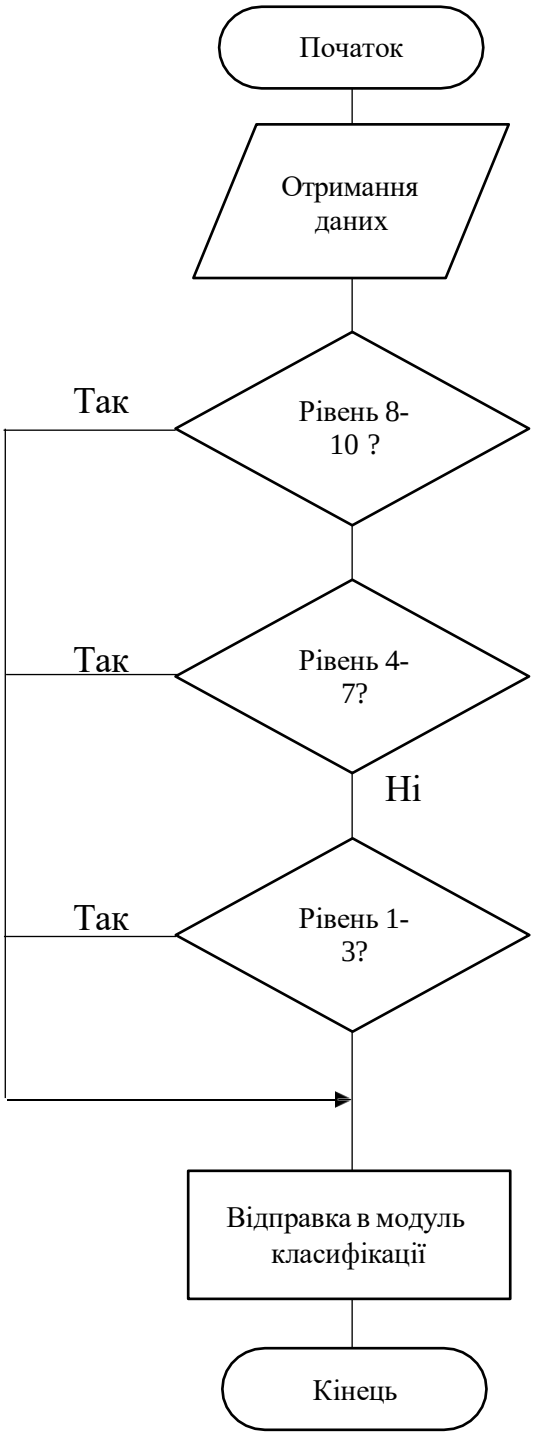


Рисунок 3.7 – Схема роботи модулю класифікації

Така система оцінювання дозволяє створити чітку шкалу для класифікації веб-ресурсів, розглядаючи різні аспекти їхнього вмісту та можливого впливу на користувачів.

3.8 Модуль звітування

Модуль звітування - це компонент програмного забезпечення, який відповідає за створення та представлення інформації про виявлений небажаний або небезпечний контент. Його основна мета - надати користувачам або адміністраторам детальну звітність щодо виявленого небезпечного вмісту для прийняття подальших заходів.

Модуль приймає результати класифікації небезпечного контенту з модулю класифікації. Проводить попередню обробку для підготовки звітів. Формує звіт та надсилає його в базу даних.

Схема роботи модулю звітування представлена на рисунку 3.8.

Рисунок 3.8 – Схема роботи модулю звітування

Модуль звітування є важливою частиною інформаційної системи, оскільки дозволяє користувачам та адміністраторам отримувати зрозумілу та цінну інформацію для прийняття рішень.

3.9 Висновки з розділу

Під час виконання розділу було розроблено архітектуру системи пошуку та аналізу небезпечного контенту інформаційних ресурсів та окремих її компонентів.

Описана архітектура системи пошуку та аналізу небезпечного контенту має ключове значення для ефективності та безпеки виявлення небажаного вмісту. Додаток включає різні модулі, які взаємодіють між собою, створюючи комплексну систему.

Після розробки архітектури система можна перейти до програмної реалізації.

4. РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ

4.1 Формування вимог до програмного засобу

Метою є розробка системи пошуку та аналізу небезпечного контенту інформаційних ресурсів. Розроблена система матиме модуль взаємодії з користувачем у вигляді веб-додатку.

Додаток повинен відповідати наступним вимогам:

- наявність графічного інтерфейсу – на основі веб-додатку;
- можливість пошуку небезпечного контенту на запропонованих користувачем сайтах або в режимі онлайн перевірятися сайти, які відвідує користувач;
- наявність роботи з файлами, для зчитування вхідної інформації;
- наявність роботи з базами даних, для зчитування та запису даних про перевірену інформацію;
- можливість аналізу інформації за вимогою користувача, а також систематичний аналіз з обраним інтервалом.

4.2 Обґрунтування засобів для реалізації.

Для реалізації програми необхідно використати одну зі сучасних об’єктно-орієнтованих мов програмування, яка працює на сервері і може відповідати на запити через HTTPS. Найпопулярнішими мовами програмування які використовуються для створення телеграм-ботів є:

- C#;
- Javascript (Node.js);
- Python.

C# - проста, сучасна мова програмування, забезпечує принципи ООП. На синтаксис мови C# має найбільший вплив C++. Ця мова є вихідним кодом під

.NET, який працює незалежно від Microsoft. C# має ряд переваг таких як:

- швидкість, порівняно з іншими мовами високого рівня;
- велика підтримка – завдяки розробникам і спільноті користувачів. Найбільша спільнота експертів знаходиться на StackOverflow, не в останню чергу через те що сайт побудований на C#;
- крос-платформа - є мовою на різних платформах. Дає змогу створювати програми .NET, які можна розгорнути не тільки на Windows, а й на Linux, Mac чи у хмарі та контейнерах;
- повністю об'єктно-орієнтований, що є рідкісною характеристикою. Багато найпоширеніших мов до певної міри включають об'єктну орієнтацію, але дуже мало хто досягнув такої величини, не втрачаючи прихильності людей. Така концепція має багато різних переваг, таких як гнучкість та ефективність;
- безпечний та ефективний - не дозволяє перетворювати типи, які призводять до точних даних або чогось іншого. Що дозволяє розробляти більш безпечний код;
- універсальний – ця мова програмування була розроблена всебічно і використовується для великої кількості різних потреб: для створення сучасних програмних додатків, використовується для розробки клієнтських програм Windows, мобільних програми, сервісів, бібліотек, веб-програм та відеоігр;
- швидко розвиваючий - 8.0 остання версія C#. Якщо поглянути на історію, то ця мова розвивається швидше, ніж будь-які інші. Завдяки Microsoft та потужній підтримці спільноти. Спочатку вона була розроблена для написання клієнтських програм Windows, але сьогодні можна робити майже все, що завгодно.

Javascript – мова сценаріїв, одна з основних мов для створення веб-сайтів. Зазвичай вона є мовою на стороні клієнта, але існує платформа Node.js, яка дозволяє виконувати код на сервері.

Зі свого трохи крихкого початку JavaScript став найпопулярнішою мовою програмування у світі. Відповідно до звіту GitHub за 2018 рік, сховищ коду на даній мові є більше, ніж будь-якої іншої, - і ця кількість постійно зростає.

JavaScript - це «безпечна» мова програмування. Вона не має низькорівневого доступ до пам'яті або центрального процесора, оскільки спочатку була реалізована тільки для браузерів, які цього не потребували. Можливості мови сильно залежать від середовища, в якому вона працює

Плюсами розробки на цій мові програмування є:

– статичний тип змінних - після оголошення вона не змінює свого типу і може приймати лише певні значення. Компілятор попереджає розробників про помилки, пов'язані з типом, тому вони не мають можливості потрапити на фазу виробництва. Це призводить до меншої кількості помилок та кращої продуктивності під час виконання;

– знаходження ранніх помилок - виявлено, що JavaScript виявляє 15 відсотків поширених помилок на етапі компіляції. Ця сума все ще є досить значною, щоб заощадити час і дозволити зосередитись на виправленні помилок в логіці;

– передбачуваність - у JavaScript все залишається так, як визначено спочатку. Якщо змінну оголошено як число, вона завжди буде цим типом і не перетвориться на щось інше. Це підвищує ймовірність функцій, що працюють так, як передбачалося спочатку;

– читабельність - завдяки строгим типам, які роблять код більш самовиразним, можна побачити задум розробників, які спочатку написали код. Це особливо важливо для команд з великою кількістю розробників, що працюють над одним проектом. Код, який говорить сам за себе, може компенсувати відсутність прямого спілкування між членами команди.

– багата підтримка IDE - інформація про типи робить редактори та інтегровані середовища розробки набагато кориснішими. Вони можуть запропонувати такі функції, як навігація кодом та автозаповнення, надаючи точні пропозиції. Ви також отримуєте зворотній зв'язок під час набору тексту: редактор позначає помилки, включаючи типи, щойно вони виникають. Все це допомагає написати код, який можна підтримувати, і призводить до значного підвищення продуктивності;

– швидкий рефакторинг - JavaScript робить цей процес менш болючим.

Оскільки середовища розробки знають багато про даний код. Крім того, багато помилок виявляються автоматично. Це спрощує та прискорює рефакторинг, що особливо корисно, коли є велика частина кодової бази;

– велика підтримка розробників - кількість розробників JavaScript сягає 11,4 мільйона - майже вдвічі більше, ніж у такий популярних мов Python або Java.

Python - це популярна мова програмування, яка має загальне призначення, що може бути використане для найрізноманітніших програм. Вона включає динамічне введення тексту та прив'язування, високорівневі структури даних та багато інших функцій, які роблять його настільки ж корисним для складної розробки додатків, як і для створення сценаріїв. Крім цього, може бути розширений для здійснення системних викликів майже до всіх операційних систем і для запуску коду, написаного на C ++. Тому Python - це універсальна мова, що зустрічається в безлічі різних застосувань.

Основними перевагами Python є:

– простий і зрозумілий синтаксис;

– відсутність дужок;

– автоматичний розподіл пам'яті;

– динамічний набір тексту;

– велика підтримка в інтернеті;

– багато підтримуваних бібліотек.

Програми чудово працюють для машинного навчання. Але Python поганий вибір для мобільних додатків.

Можливі мінуси Python:

– повільний і може стати громіздким для великих і складних програм;

– мова високого рівня, яка не підходить для написання програмних програм;

– для деяких завдань неявне виділення пам’яті може бути недоліком.

Для реалізації програмного засобу обрано мову програмування Javascript, оскільки ця мова має такі переваги:

- незалежність від платформи, на якій виконується програма;
- мова є повністю об’єктно-орієнтованою;
- повна інкапсуляція;
- відсутність глобальних функцій;
- зручні бібліотеки для виконання специфічних завдань бота.

Порівняння мов програмування і платформ між собою показало, що Javascript та платформа Node.js мають сучасні переваги перед своїми конкурентами в рамках розробки даного програмного застосунку.

Крім того, Javascript ідеально підходить для роботи з TelegramBotAPI та TelegramSession – ключовими модулями для роботи телеграм-боту.

Серед таких середовищ, як Atom, Webstorm, Visual Studio Code, в межах робити обрано Visual Studio Code. Він є безкоштовним, має інтеграцію з Git, і режим налагодження коду. Підтримує мову програмування Javascript, автоматично виділяє синтаксичні конструкції, має підказки та довідку.

Реалізація графічного інтерфейсу застосунку відбулась у вигляді веб-розширення для браузера Google Chrome.

4.3 Реалізація графічного інтерфейсу системи

Для графічного інтерфейсу системи обрано реалізацію у вигляді веб-розширення для браузеру Google Chrome.

На рисунку 4.1 зображено вигляд веб-розширення після завантаження з магазину веб-розширень Google Chrome.

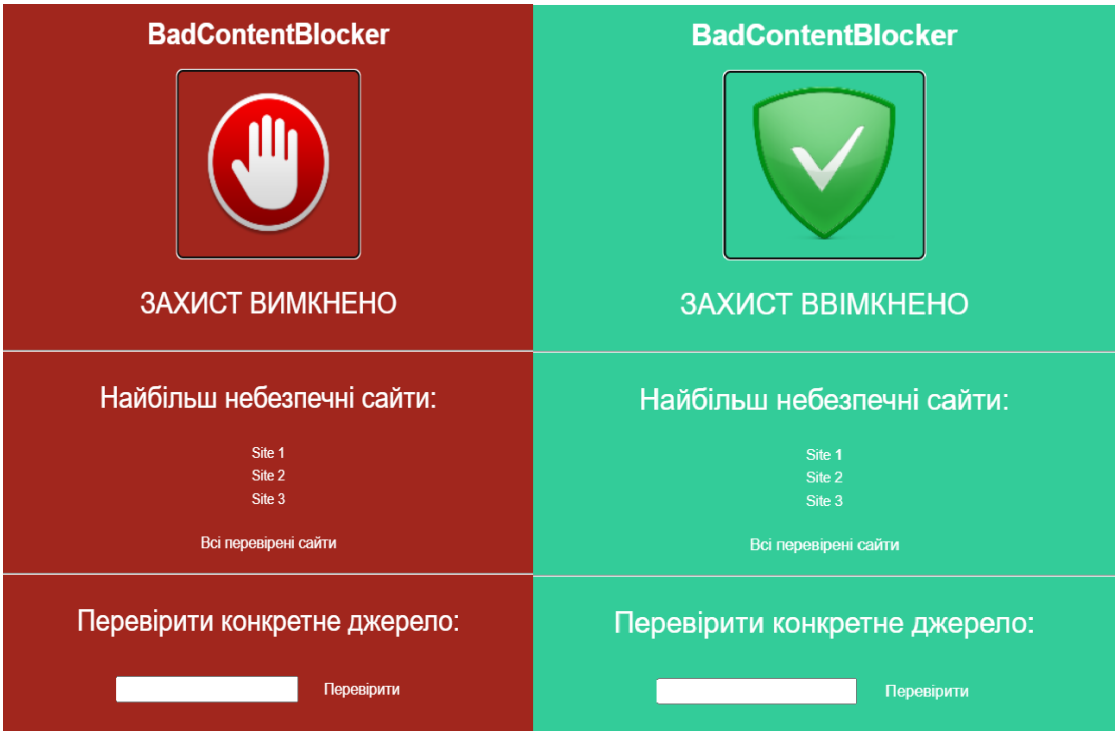


Рисунок 4.1 – Вигляд веб-розширення одразу після завантаження

В головному меню розміщено кнопку для ввімкнення онлайн захисту у вигляді рисунка руки. При натисканні фон розширення і рисунок кнопки змінюється.

Крім того, нижче розміщено повідомлення про статус онлайн захисту, список найбільш небезпечних сайтів з усіх перевірених системою в даного користувача.

Також система може працювати в режимі перевірки певного джерела, з цією метою нижче розміщено поле для вводу посилання на сайт, а також кнопка для підтвердження.

Для отримання повного звіту по всіх перевірених сайтах є клікабельне посилання «Всі перевірені сайти», при натисканні на яке відбувається завантаження csv-файлу з усією статистикою перевірок.

4.4 Налаштування роботи з базою даних.

Для роботи засобу і усунення зайвих перевірок одного сайту по багато разів, застосунок повинен працювати з базою даних, куди записувати знайдений та

проаналізований небезпечний контент сайт де його знайдено і читати її перед перевіркою, якщо результат вже є – виводити його.

Для програмного застосунку обрано систему керування базами даних MySql. MySQL відома своєю високою продуктивністю та швидкодією[22]. Вона працює ефективно навіть з великими обсягами даних, що робить її ідеальним вибором для додатків з вимогами до швидкодії та відповідності.

Можливості MySQL:

- висока швидкість;
- підтримка реплікації;
- гнучкість використання;
- використання індексів;
- підтримка транзакцій та замків;
- використання сховищ та процедур.

Для роботи з телеграм-ботом, спочатку потрібно створити базу даних, це можна зробити через веб-інтерфейс phpMyAdmin[23] або через програму MySQL Workbench.

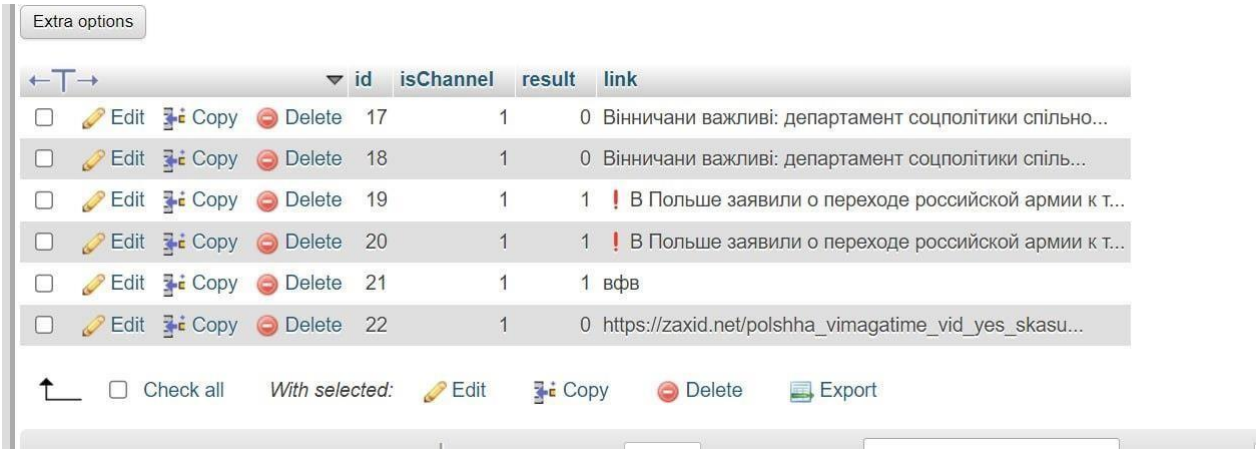


Рисунок 4.3 – База даних для роботи системи

На рисунку вище зображено готову до використання базу даних MySQL, далі збережене посилання для підключення і пароль потрібно вписати в файлі config.js.

На рисунку 4.4 зображено вміст файлу config.js, в якому містяться потрібні для роботи засобу токени.

```
require('dotenv').config()

const databaseConfig = {
  host: process.env.DATABASE_HOST,
  user: process.env.DATABASE_USER,
  database: process.env.DATABASE_NAME,
  password: process.env.DATABASE_PASSWORD,
};

module.exports = databaseConfig;
```

Рисунок 4.4 – Вміст файлу config.js

Для створення з’єднання і можливості читання/запису в базу даних в застосунку використовується об’єктно орієнтована бібліотека dotenv [24]. Під час запуску телеграм-боту через термінал VS Code, вкінці додано вивід в термінал “DB has connected” що означає, що все функціонує правильно і база даних підключена і готова до запису чи читання.

4.5 Реалізація модулю парсингу

Парсинг системи виконується у вигляді веб-скрапінгу.

Для веб-скрапінгу використовуються бібліотеки axios[25] та cheerio[26]. За допомогою методів бібліотеки axios відправляється get-запит з посилання користувача, а за допомогою методів бібліотеки cheerio відбувається завантаження контенту з веб-сторінки користувача в спеціально відведену для цього змінну.

Парсинг тексту ресурсу за адресою зображено на рисунку 4.5.

```
const axios = require('axios');
const cheerio = require('cheerio');

const url = user.send.url;

axios.get(url)
  .then(response => {
    const html = response.data;
    const $ = cheerio.load(html);

    // Отримання тексту
    const text = $('body').text();
    console.log('Текст сторінки:', text);
```

Рисунок 4.5 – Парсинг тексту за адресою користувача

В разі веб-скрапингу картинок чи іншого графічного контенту відбувається зчитування тексту з картинки за допомогою бібліотеки Tesseract.js.

Бібліотека може бути корисна при зчитуванні тексту з фото, візиток, документів. Крім того, крім тексту повертається «рівень впевненості», який можна використовувати для точнішого перекладу та аналізу тексту. Проте, якщо контент має багато деталей або набраний нестандартними шрифтам рівень розпізнавання може суттєво впасти.

Функція для розпізнавання тексту з картинки зображена на рисунку 4.6.

```
function recognize(file, lang, logger) {
  return Tesseract.recognize(file, lang, {logger})
    .then(({ data: {text }}) => {
      return text;
    })
}
```

Рисунок 4.6 – Функція для розпізнавання тексту з картинки

У випадку активної онлайн перевірки веб-розширення, отримує посилання з заголовку файлів браузера користувача і проводить веб-скрапінг за цим посилання отримуючи текст та графічний контент для аналізу.

4.6 Реалізація модулю аналізу

Для реалізації модулю аналізу використовується бібліотека natural[27].

Спочатку над отриманим після парсингу текстом потрібно провести лексичний аналіз, так званий процес токенізації тексту.

За допомогою бібліотеки natural, а саме методу WordTokenizer, з переданої стрічки тексту на виході отримуємо слова розбиті на окремі токени для послідуючого аналізу.

Токенізація тексту представлена на рисунку 4.7.

```
var natural = require('natural');
var tokenizer = new natural.WordTokenizer();
```

Рисунок 4.7 – Токенізація тексту

Після токенізації відбувається стеммінг тексту - зведення слів до їх основ. Natural має вбудований метод для стеммінгу на основі методу стеммера Портера[27] (рис. 4.8).

```
var natural = require('natural');

natural.PorterStemmer.attach();
```

Рисунок 4.8 – Стеммінг токенів

Для стеммінгу використовується метод attach() котрий патчить методи stem() і tokenizeAndStem() до string, будучи скороченням до PorterStemmer.stem(token)tokenizeAndStem(). Результатом ємасив згенерованих токенів-основ слів, розбитих на токени на попередньому етапі. Крім-того, алгоритм самостійно видаляє з тексту шумові та стоп-слова.

Аналіз тональності тексту (також відомий як opinion mining або emotion AI) застосовується в обробці природної мови, аналізі текстових даних, комп'ютерній

лінгвістиці та біометрії для систематичної ідентифікації, вилучення, кількісної оцінки та вивчення афективних станів та суб'єктивної інформації.

Natural підтримує алгоритми, які можуть обчислити емоційне забарвлення кожного фрагмента тексту, підсумовуючи полярність кожного слова та нормалізуючи його за довжиною речення (рис. 4.9).

```
var natural = require('natural');
var Analyzer = natural.SentimentAnalyzer;
var stemmer = natural.PorterStemmer;
var analyzer = new Analyzer("Ukrainian", stemmer, "afinn");
```

Рисунок 4.9 – Аналіз тональності

Конструктор приймає три параметри – мова, стеммер (передається для збільшення охвату аналізатора), словник – встановлює тип словник і має три можливі значення.

4.7 Реалізація модулю звітування

Модуль звітування працює за принципом створення csv-файлу з записаними в неї результатами всіх перевірок та їх результатів користувача.

В графічному інтерфейсі користувача це зображено у вигляді клікабельного посилання на всі перевірки, в разі натискання на яке відбувається завантаження файлу (рис. 4.10)

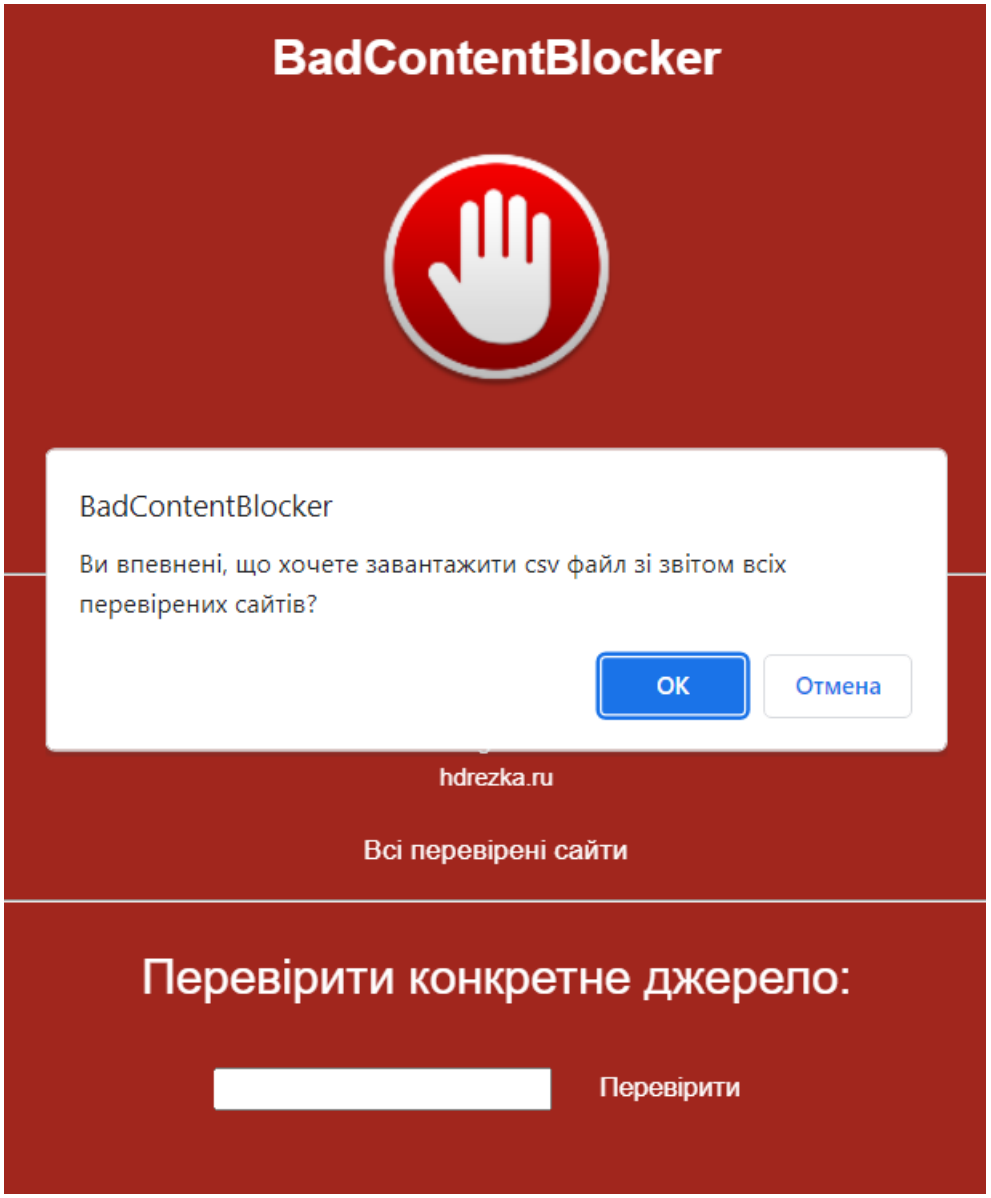


Рисунок 4.10 – Завантаження звіту користувачем

З метою звітування користувачі про всі перевірки, було прийнято рішення реалізувати в системі csvmaker.

Після підтвердження завантаження користувач отримує таблицю зі всіма результатами перевірки.

Файл звіту представлено на рисунку 4.12.

	A	B	C	D	E	F
1	name	link	result	type	level	
2	Hdrezka	hdrezka.ru	1	азартні ігри	5	
3	Omegle	omegle.com	1	кадри оголення	8	
4	Instagram	instagram.cc	0	-	0	
5						

Рисунок 4.11 – Таблиця зі звітом

Таблиця містить такі поля як : ім’я, посилання, результат перевірки, тип небезпечного контенту, якщо такий знайдено, та рівень небезпечності сайту.

4.8 Тестування роботи системи в режимі перевірки конкретного джерела.

З метою тестування роботи системи в режимі перевірки конкретного джерела було прийнято рішення протестувати систему в реальній роботі та перевірити кілька сайтів за посиланнями і переглянути звіт.

Першим сайтом для перевірки обрано сайт з товаром з інтернет-магазину.

На обраному для перевірки сайті знаходиться один з товарів інтернет-магазину, а саме блютуз-гарнітура (рис. 4.12).

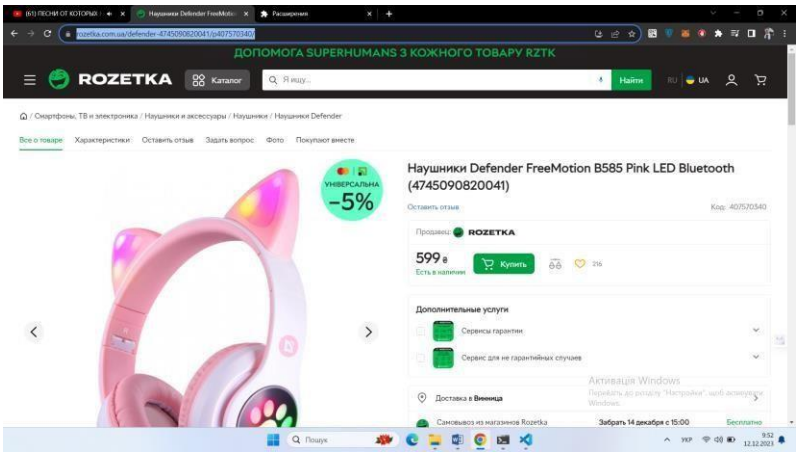


Рисунок 4.12 – Вміст сайту для тестування

Отже, сайт не містить жодного небезпечного контенту і система має це розпізнати.

Для перевірки скопійоване посилання потрібно вставити в поле вводу

користувача і підтвердити перевірку (рис. 4.13).

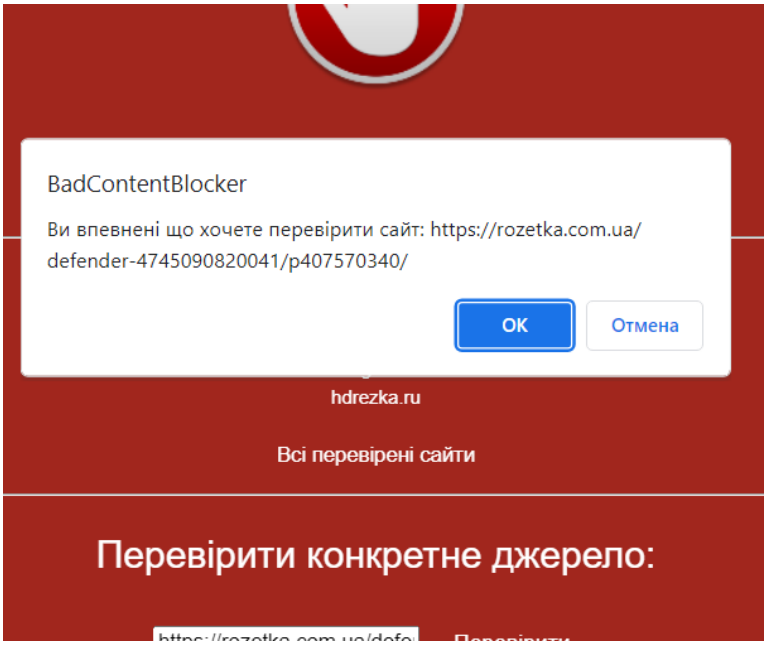


Рисунок 4.13 – Перевірка сайту

Після недовгого часу перевірки система відправляє повідомлення про результат, а також робиться запис в базі даних та в таблиці, яку може завантажити користувач.

Повідомлення користувачу про результат перевірки зображено на рисунку 4.14.

Рисунок 4.14 – Повідомлення користувачу про результат перевірки

Система правильно визначила, що даному сайті не міститься небезпечного контенту та він є повністю безпечним.

Друге джерело для перевірки має містити небезпечний контент для тестування його виявлення системою. З цією метою обрано новинний ресурс Луцьку, і його скандальну статтю про ромів, в якій психологи помічають мову ворожнечі та розпалювання міжнаціональної ворожнечі (рис.4.15).

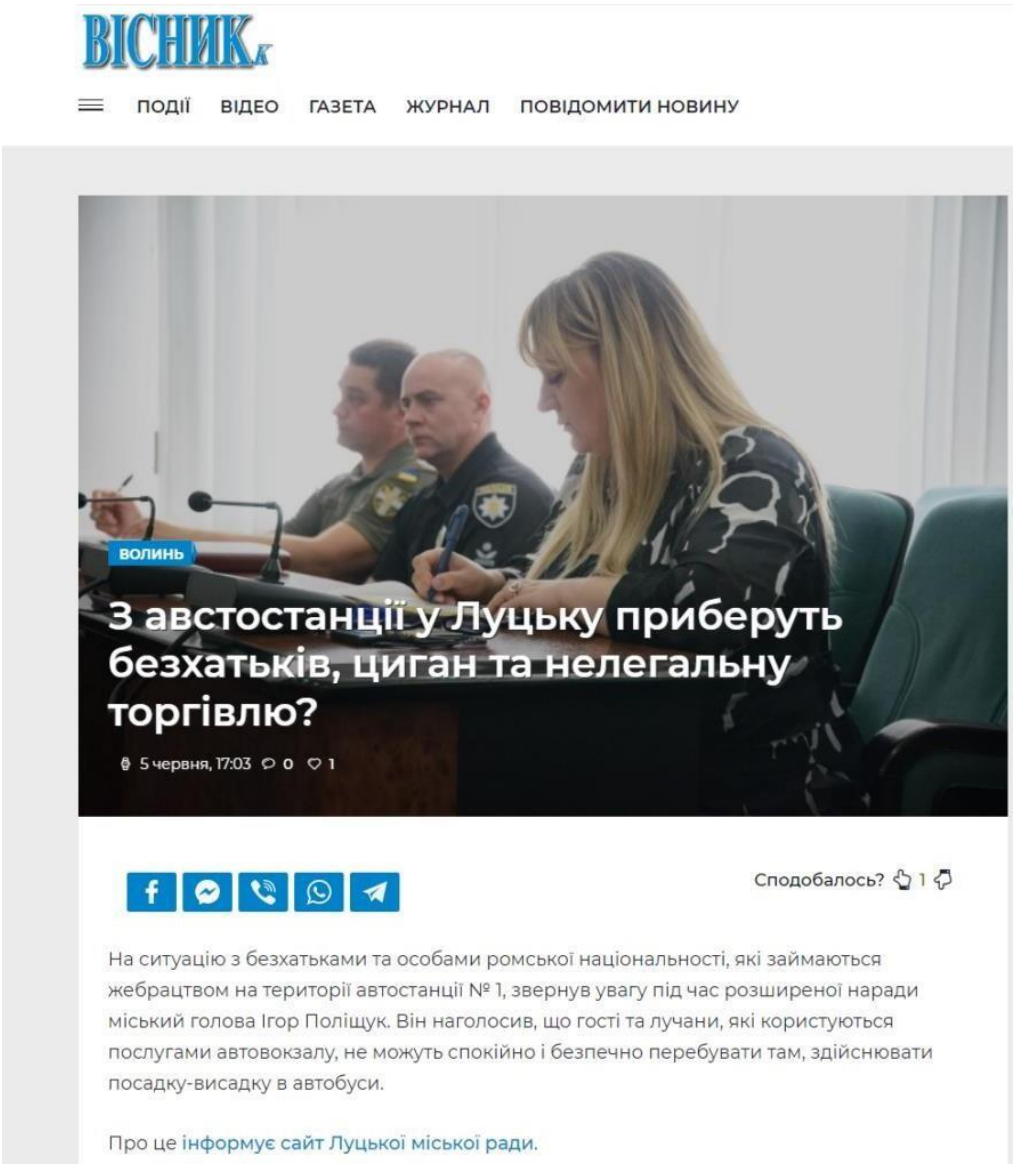


Рисунок 4.15 – Вміст сайту для перевірки

Експерти зазначають, що в заголовку даної новини видно ознаки мови ворожнечі, розпалювання ненависті та порушення журналістської етики, саме це спробує визначити розроблена система.

Результат перевірки новинного ресурсу зображено на рисунку 4.16.

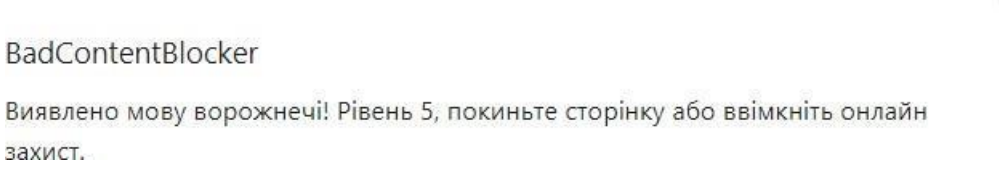


Рисунок 4.16 – Результат перевірки

Отже, система показала коректну роботу в режимі перевірки конкретного джерела.

4.9 Тестування роботи системи в режимі онлайн-захисту.

В режимі онлайн-захисту користувач вмикає його за допомогою кнопки та бачить візуальні зміни у вигляді розширення, це означає що захист увімкнуто і він працює (рис. 4.17).

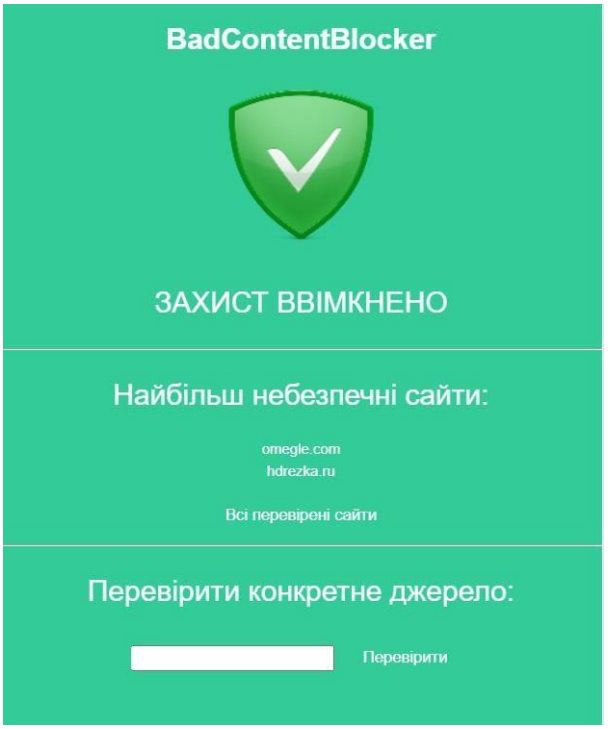


Рисунок 4.17 – Вигляд розширення при активованому режимі онлайн-захисту

Після цього користувач може спокійно серфити веб-сторінки, система буде в онлайн-режимі моніторити відвідувані джерела, блокувати доступ до сторінок, на яких виявлено небезпечний контент та робити записи в базу даних та таблицю для звітування.

Після моніторингу сторінки, що містить небезпечний контент, доступ до блокується а користувач отримує замість неї html-файл з повідомленням про

загрозу (рис. 4.18).

Таким чином відбувається усунення негативного впливу небезпечного контенту на користувача.

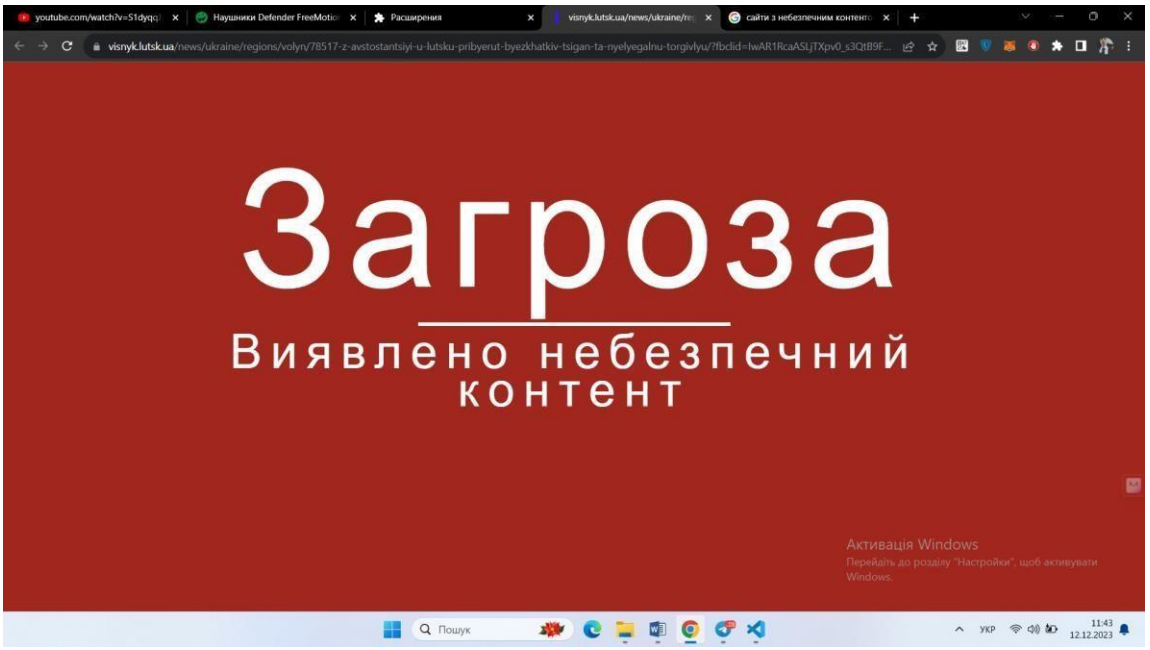


Рисунок 4.18 – Робота режиму онлайн-захисту

Таким чином, можна захистити користувачів від впливу небезпечного контенту виявленого і проаналізованого системою.

Отже, система показала гарні показники в виявленні та аналізі небезпечного контенту, а також швидкість роботи та реагування.

Більше детальна статистика тестування системи на реальних прикладах представлена в таблиці 4.1.

Таблиця 4.1 – Статистика тестування системи

Перевірені сайти :		Визначення системи				Коректність роботи
		Рівень 0	Рівень 1-3	Рівень 4-7	Рівень 8-10	
Всього:	300					

З небезпечним контентом	150	13	69	47	21	80,5%
Без небезпечного контенту	150	131	19	0	0	71,5%

Для проведення масштабного тестування роботи системи було перевірено 150 веб-сайтів з небезпечним контентом і стільки ж без нього.

Система показала гарні результати з правильності класифікації небезпечного контенту по рівням та порівняно низькі показники хибних спрацювань.

Загальний процент коректності роботи системи склав 76%, що вважається гарним результатом для систем такого рівня.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів" є покращення кібербезпеки шляхом створення системи для пошуку та аналізу небезпечного контенту.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 5.1 [29].

Таблиця 5.1 – 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 властивості 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
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років

12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь- які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці. Для оцінки науково-технічного рівня і комерційного потенціалу розробки експертами було запрошено трьох незалежних експертів: Олеся Петрівна Войтович к.т.н. доцент кафедри захисту інформації Вінницького національного технічного університету, Олексій Палій Middle Software Engineer ТОВ «СМІСС», Дмитро Поворозник спеціаліст з налаштування інформаційних систем захисту ТОВ «АТК»

Таблиця 5.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	Олеся Войтович к.т.н. доц.каф. ЗІ	Олексій Палій Middle Software Engineer ТОВ «СМІСС»	Дмитро Поворозник спеціаліст з налаштування інформаційних систем захисту ТОВ «АТК»
	Бали, виставлені експертами:		
1. Технічна здійсненність концепції	4	4	4
2. Ринкові переваги (наявність аналогів)	4	4	4
3. Ринкові переваги (ціна продукту)	2	2	2
4. Ринкові переваги (технічні властивості)	4	3	4
5. Ринкові переваги (експлуатаційні витрати)	3	3	3
6. Ринкові перспективи (розмір ринку)	2	3	3

7. Ринкові перспективи (конкуренція)	2	3	3
8. Практична здійсненність (наявність фахівців)	3	3	3
9. Практична здійсненність (наявність фінансів)	3	3	3
10. Практична здійсненність (необхідність нових матеріалів)	3	3	3
11. Практична здійсненність (термін реалізації)	4	4	4
12. Практична здійсненність (розробка документів)	2	2	2
Сума балів	$СБ_1$=31	$СБ_2$=34	$СБ_3$=34
Середньоарифметична сума балів $СБ_c$	$\overline{СБ} = \frac{\sum_{1}^{3} СБ_i}{3} = \frac{31+34+34}{3} = 33$		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 5.3[29].

Таблиця 5.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ , розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41…48	Високий
31…40	Вище середнього
21…30	Середній
11…20	Нижче середнього
0…10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів" становить 33 бали, що, відповідно до таблиці 4.3 рівень комерційного потенціалу розробки вище середнього, що свідчить про комерційну важливість проведення даних досліджень.

Магістерська кваліфікаційна робота "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів" відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок, тобто при цьому відбувається комерціалізація

науково-технічної розробки. Цей напрямок є для нас пріоритетним, оскільки результатами розробки можуть користуватися не тільки самі розробники, а й інші споживачі, отримуючи при цьому суттєвий економічний ефект.

Результатом магістерської роботи є програмний засіб у вигляді веб-розширення для пошуку та аналізу небезпечного контенту, який може бути корисним для користувачів веб-браузера Google Chrome.

5.2 Визначення рівня конкурентоспроможності розробки

В процесі визначення економічної ефективності науково-технічної розробки також доцільно провести прогноз рівня її конкурентоспроможності за сукупністю параметрів, що підлягають оцінюванню.

В якості аналога для розробки було обрано WOT. Основними недоліками аналога є відсутність онлайн захисту. Також до недоліків можна віднести відсутність класифікації небезпечного контенту.

У розробці дана проблема вирішується за допомогою розробки та реалізації власної системи класифікації небезпечного контенту, та запровадження онлайн захисту. Також система випереджає аналог за такими параметрами як швидкість роботи.

Одиничний параметричний індекс розраховуємо за формулою [29].

$$q_i = \frac{P_i}{P_{базі}}. \quad (5.1)$$

де q_i – одиничний параметричний індекс, розрахований за i-м параметром;

P_i – значення i-го параметра виробу;

$P_{базі}$ – аналогічний параметр базового виробу-аналога, з яким проводиться порівняння.

Загальні технічні та економічні характеристики розробки представлено в таблиці 5.4.

Таблиця 5.4 – Основні техніко-економічні показники аналога та розробки, що проектується

Показник	Варіанти		Відносний показник якості	Коефіцієнт вагомості параметра
	Базовий (товар-конкурент)	Новий (інноваційне рішення)		
1	2	3	4	5
Доступність сервісу,%	98	100	1,02	20 %
Швидкодія сервісу, мс	150	120	1,25	10 %
Вартість обслуговування на одного користувача, грн	35	5	7	20 %
Кількість виявлених врахливостей, шт	10	3	3,33	30 %
Час повної перевірки, хв	1	0.5	2	20 %

Нормативні параметри оцінюємо показником, який отримує одне з двох значень: 1 – пристрій відповідає нормам і стандартам; 0 – не відповідає.

Груповий показник конкурентоспроможності за нормативними параметрами розраховуємо як добуток частинних показників за кожним параметром за формулою [29] :

$$I_{НП} = \prod_{i=1}^{n} q_i \ , \tag{5.2}$$

де $I_{нп}$ – загальний показник конкурентоспроможності за нормативними параметрами;

q_i – одиничний (частинний) показник за i-м нормативним параметром;

n – кількість нормативних параметрів, які підлягають оцінюванню.

За нормативними параметрами розроблюваний пристрій відповідає вимогам ДСТУ, тому $I_{нп} = 1$.

Значення групового параметричного індексу за технічними параметрами визначаємо з урахуванням вагомості (частки) кожного параметра:

$$I_{ТП} = \sum_{i=1}^{n} q_i \cdot \alpha_i \ , \tag{5.3}$$

де $I_{тп}$ – груповий параметричний індекс за технічними показниками (порівняно з виробом-аналогом);

q_i – одиничний параметричний показник i-го параметра;

α_i – вагомість i-го параметричного показника, $\sum_{i=1}^{n}\alpha_i = 1$;

n – кількість технічних параметрів, за якими оцінюється конкурентоспроможність.

Проведемо аналіз параметрів згідно даних таблиці 4.4.

$$I_{mn} = 1{,}02\cdot 0{,}2+1{,}25\cdot 0{,}1+7\cdot 0{,}2+3{,}33\cdot 0{,}3+2\cdot 0{,}2= 3{,}13.$$

Груповий параметричний індекс за економічними параметрами розраховуємо за формулою [29]:

$$I_{ЕП} = \sum_{i=1}^{m} q_i \cdot \beta_i \,, \qquad (5.4)$$

де $I_{ЕП}$ – груповий параметричний індекс за економічними показниками;

q_i – економічний параметр i-го виду;

β_i – частка i-го економічного параметра, $\sum_{i=1}^{m}\beta_i = 1$;

m – кількість економічних параметрів, за якими здійснюється оцінювання.

Проведемо аналіз параметрів згідно даних таблиці .

$$I_{ЕП}=0{,}75 \cdot 0{,}5 + 0{,}86 \cdot 0{,}5 = 0{,}80.$$

На основі групових параметричних індексів за нормативними, технічними та економічними показниками розрахуємо інтегральний показник конкурентоспроможності за формулою **:**

$$K_{ІНТ} = I_{НП} \cdot \frac{I_{ТП}}{I_{ЕП}}, \qquad (5.5)$$

$$K_{ІНТ} = 1 \cdot 3{,}13 / 0{,}80 = 4.$$

Інтегральний показник конкурентоспроможності $К_{ІНТ} > 1$, отже розробка переважає відомі аналоги за своїми техніко-економічними показниками.

5.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов’язані з проведенням науково-дослідної роботи на тему "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів", під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

5.3.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників ($З_о$) розраховуємо у відповідності до посадових окладів працівників, за формулою :

$$З_о = \sum_{i=1}^{k} \frac{M_{ni} \cdot t_i}{T_p}, \quad (5.6)$$

де k – кількість посад дослідників залучених до процесу досліджень;

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

t_i – число днів роботи конкретного дослідника, дн.;

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

$$З_о = 18000 \cdot 5 / 21 = 4091 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 5.5 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	18000	818,2	5	4091
Програміст	13000	590,9	42	24818
Всього				28909

Основна заробітна плата робітників

Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт НДР на тему "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів" розраховуємо за формулою:

$$З_р = \sum_{i=1}^{n} C_i \cdot t_i \,, \tag{5.7}$$

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

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

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

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}} \,, \tag{5.8}$$

де M_M – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), приймемо $Мм$=6500 грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б)**;**

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

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

$t_{зм}$ – тривалість зміни, год.

$С_1$ = 6700,00 · 1 · 1,65 / (21 · 8) = 65,8 грн.
$З_{р1}$ = 65,8 · 1 = 65,8 грн.

Таблиця 5.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
1.Підготовчі	18	1	65,8	1184,5
2.Монтажні	34	3	88,8	3020,4
3.Складальні	35	5	111,9	3915,3
4.Налагоджувальні	38	2	72,4	2750,6
5.Випробувальні	27	4	59,8	1615,2
Всього				12485,9

Додаткова заробітна плата дослідників та робітників

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

$$З_{дод} = (З_о + З_р) \cdot \frac{Н_{дод}}{100\%}, \quad (5.9)$$

де $Н_{дод}$ – норма нарахування додаткової заробітної плати. Приймемо 11%.

$З_{дод}$ = (28909+12485,9) · 11 / 100% = 4553,45 грн.

5.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_н = (З_о + З_р + З_{дод}) \cdot \frac{Н_{зп}}{100\%} \quad (5.10)$$

де $Н_{зп}$ – норма нарахування на заробітну плату. Приймаємо 22%.

$Зн$ = (28909+12485,9+553,45) · 22 / 100% = 10108,66 грн.

5.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення

досліджень за темою "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів".

Витрати на матеріали (*М*), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^{n} H_j \cdot Ц_j \cdot K_j \; - \sum_{j=1}^{n} B_j \cdot Ц_{в\,j} \;, \qquad (5.11)$$

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

n – кількість видів матеріалів;

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

K_j – коефіцієнт транспортних витрат, (K_j = *1,1 ... 1,15*);

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

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

Проведені розрахунки зведемо до таблиці.

Таблиця 5.7 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Вартість витраченого матеріалу, грн
Папір канцелярський офісний (А4)	170	1	170
ручка	50	1	50
Флешка	250	1	250
Всього			470
З врахуванням коефіцієнта транспортування			517

5.3.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі *($К_в$)*, які використовують при проведенні НДР на тему "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів".

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

$$К = \sum_{1}^{n} Н_i \cdot Ц_i \cdot К_i \text{ грн.,} \quad (5.12)$$

де $Н_i$ – кількість комплектуючих *i*-го виду, шт.;

$Ц_i$ – ціна комплектуючих *i*-го виду, грн.;

$К_i$ – коефіцієнт транспортних витрат, $К_i = (1{,}1 \ldots 1{,}15)$;

n – кількість видів комплектуючих.

Зроблені розрахунки бажано звести до таблиці:

Таблиця 5.8 – Витрати на комлектуючі

Найменування комплектувальних	Кількість	Ціна за штуку, грн.	Сума, грн.
Intel Xeon E5-2620, 64 ГБ оперативної пам'яті, 1 ТБ SSD	2	100000	200000
PostgreSQL	1	15000	15000
Мереже обладнання	1	10000	10000
DataStorage	4	60000	240000
Всього з врахування коефіцієнт транспортних витрат			511500,00

5.3.5 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_в} \cdot \frac{t_{вик}}{12}, \quad (5.13)$$

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

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

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

$A_{обл} = (28000 \cdot 2) / (2 \cdot 12) = 2333{,}33$ *грн.*

Проведені розрахунки зведемо до таблиці.

Таблиця 5.9 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Ноутбук Lenovo Legion	28000	2	2	2333,33
Робоче місце розробника ПЗ	220000	20	3	2750,00
Всього				5083,33

5.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

Результати дослідження проведені за темою "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів" передбачають комерціалізацію протягом 3-х років реалізації на ринку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

ΔN – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються;

N – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, приймемо 1 особа

$Ц_б$ – вартість послуги у році до впровадження інформаційної системи, приймемо 400,00 грн;

$\pm\Delta Ц_o$ – зміна вартості послуги від впровадження результатів, приймемо зростання на 50,00 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta П_i$ для кожного із 3-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою **:**

$$\Delta П_i = (\pm\Delta Ц_o \cdot N + Ц_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\vartheta}{100}), \quad (5.20)$$

де λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2023 році ставка податку на додану вартість складає 20%, а коефіцієнт λ *=0,8333*;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту). Приймемо ρ =40*%*;

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

Збільшення чистого прибутку 1-го року:

$\Delta П_1 = (1 \cdot 50 + 400 \cdot 18000) \cdot 0{,}83 \cdot 0{,}4 \cdot (1\text{-}0{,}18/100\%) = 1383703{,}2$ грн.

Збільшення чистого прибутку 2-го року:

$\Delta П_2 = (1 \cdot 50 + 400 \cdot (18000 + 14000)) \cdot 0{,}83 \cdot 0{,}4 \cdot (1\text{-}0{,}18/100\%) = 2459951{,}6$ грн.

Збільшення чистого прибутку 3-го року:

$\Delta П_3 = (1 \cdot 50 + 400 \cdot (18000 + 14000 + 11000)) \cdot 0{,}83 \cdot 0{,}4 \cdot (1\text{-}0{,}18/100\%) = 3305542{,}8$ грн.

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

$$ПП = \sum_{i=1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \quad (5.21)$$

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

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

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, τ =18%;

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

$$ПП = 1383703{,}2 /(1+0{,}18)^1+2459951{,}6 /(1+0{,}18)^2+3305542{,}8 /(1+0{,}18)^3=$$
$$= 4783212{,}91 \text{ грн.}$$

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

$$PV = k_{інв} \cdot ЗВ, \quad (5.22)$$

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{інв}$=2;

$ЗВ$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 715441,08 грн.

$PV = k_{інв} \cdot ЗВ = 2 \cdot 715441{,}08 = 1430882{,}16$ грн.

Абсолютний економічний ефект $E_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV \quad (5.23)$$

де $ПП$ – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 4783212,91 грн;

PV – теперішня вартість початкових інвестицій, 1430882,16 грн.

$E_{абс} = ПП - PV$ = 4783212,91 - 1430882,16 =3352330,75 грн.

Внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1, \quad (5.24)$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, грн;

PV – теперішня вартість початкових інвестицій, грн;

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

$$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1 = (1+= 3352330,75 /1430882,16)^{1/3} - 1 = 0,78.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{мін}$:

$$\tau_{мін} = d + f, \quad (5.25)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні d =0,1;

f – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,25.

$\tau_{мін}$ = 0,1+0,25 = 0,35 < 0,78 свідчить про те, що внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою

«Інформаційна технологія онтологічного моделювання бази знань з організації бібліотеки» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_{в}}, \quad (5.26)$$

де $E_{в}$ – внутрішня економічна дохідність вкладених інвестицій.

$T_{ок}$ = 1 / 0,78 = 1,3 р.

$T_{ок}$ < 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

5.5 Висновки до розділу

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів" становить 33 бали, що, свідчить про комерційну важливість проведення даних досліджень оскільки рівень комерційного потенціалу розробки вище середнього.

При оцінюванні рівня конкурентоспроможності, згідно узагальненого коефіцієнту конкурентоспроможності розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 4 рази.

Також термін окупності становить 1,3 роки, що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою "Система пошуку та аналізу небезпечного контенту інформаційних ресурсів".

ВИСНОВКИ

Результатом виконання магістерської кваліфікаційної роботи є система пошуку та аналізу небезпечного контенту інформаційних ресурсів. За допомогою розробленого засобу перевірено різні веб-сайти на вміст небезпечного контенту.

Для пошуку та аналізу небезпечного контенту використовується веб-скрапінг, зчитування тексту з візуального контенту, стеммінг, а також семантичний аналіз тексту та моделі обробки природньої мови.

Крім того, веб-розширення має функцію онлайн моніторингу веб-серфингу користувача, котрий забезпечує захист користувача від впливу небезпечного контенту шляхом блокування доступу до веб-сторінок з ним.

Система показала непоганий результат по пошуку та аналізу небезпечного контенту інформаційних ресурсів на реальних даних.

Можна зробити висновок, що використання збору інформації з веб-сайтів, з її подальшим аналізом та дослідженням є перспективним методом в покращенні кібербезпеки окремих користувачів та суспільства в цілому. Також це допоможе покращити інформаційну безпеку користувачів, їх психологічне здоров’я та вберегтися від впливу небезпечного контенту інформаційних ресурсів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Поляков О. М. Особливості протидії поширенню деструктивного контенту //Інформація і право. – 2023. – №. 1 (44). – С. 129-141.

2. Оксана Т. НЕГАТИВНІ ПРАКТИКИ ЯК РЕАКЦІЯ НА ДЕСТРУКТИВНИЙ КОНТЕНТ //The 17th International scientific and practical conference “System analysis and intelligent systems for management”(May 02–05, 2023) Ankara, Turkey. International Science Group. 2023. 482 p. – 2023. – С. 52.

3. Троцька, Аліна. Використання методів пошукової оптимізації сайтів у мережі Інтернет. BS thesis. КПІ ім. Ігоря Сікорського, 2022.

4. Руслан П’ятак. Система пошуку та аналізу небезпечного контенту інформаційних ресурсів. Міжнародна науково-практична інтернет-конференція студентів, аспірантів та молодих науковців «Молодь в науці : дослідження, проблеми, перспективи» ВНТУ, 2023, 2с. URL: https://conferences.vntu.edu.ua /index.php/mn/mn2024/paper/view/19728 (дата звернення 10.12.2023)

5. The Budapest Convention (ETS No. 185) and its Protocols. URL: https://www.coe.int/en/web/cybercrime/the-budapest-convention (дата звернення 16.10.2023)

6. Лубенець, І. Г. "RISKS AND THREATS TO CHILDREN IN THE DIGITAL ENVIRONMENT AS A SUBJECT OF CRIMINOLOGICAL RESEARCH РИЗИКИ ТА ЗАГРОЗИ ДЛЯ ДІТЕЙ В ЦИФРОВОМУ СЕРЕДОВИЩІ ЯК ПРЕДМЕТ КРИМІНОЛОГІЧНОГО." INFORMATION TECHNOLOGIES AND MANAGEMENT IN HIGHER EDUCATION AND SCIENCES (2022): 216.

7. Bousnane R. Safe search engines to protect children and negative digital content Reality and rationalization mechanisms. 2022. URL: http://dspace.univ-batna.dz/xmlui/handle/123456789/4343 (дата звернення 22.10.2023)

8. Огляд фільтру SmartScreen Microsoft 2023. URL: https://learn.microsoft.com/ru-ru/windows/security/operating-system-security/virus-and-threat-protection/microsoft-defender-smartscreen/ (дата звернення 23.10.2023)

9. About OpenDNS 2022. URL: https://www.opendns.com/about/ (дата звернення 27.10.2023)

10. Захист мережі з Symantec WebFilter URL: http://www.infobezpeka.com/products/trafic/Symantec_Web_Gateway/ (accessed: 29.10.2023)

11. Content Keeper Solutions URL: https://www.contentkeeper.com/solution/cloud-security (дата звернення: 15.10.2023)

12. Khan, Md Ashraful Azam, Haslinda Hashim, and Lee WeiYing. "Effect of Perceived Risks, Perceived Benefits, Products Trust and Web-Vendor Trust on Online Purchase Intention of Green Personal Care Product among Gen Y in Malaysia." Journal of International Business and Management 6.3 (2023): 01-15.

13. Norton Family Products URL : https://ru.norton.com/products/norton-family (дата звернення 29.10.2023)

14. Таранова, Тетяна Андріївна. Система аналізу тональності тексту з метою виявлення токсичних коментарів. BS thesis. КПІ ім. Ігоря Сікорського, 2021.

15. Сітко, А. В., І. В. Струк, and Г. Г. Єнчева. "Контрастивний аналіз синтаксичної структури англійської, української та китайської мов." Китаєзнавчі дослідження 2 (2021): 136-147.

16. Кудрявцев, А. М. "Інформаційна технологія екстракції метаданих для аналізу документів." (2020).

17. Scrapy Documentation URL : https://docs.scrapy.org/ (дата звернення 24.11.2023)

18. Requests.js Documentation URL : https://github.com/request/request (дата звернення 24.11.2023)

19. Мушта Ілля Андрійович, Ілля Андрійович. Методи фільтрації хмари точок отриманої з Lidar сенсору. MS thesis. КПІ ім. Ігоря Сікорського, 2022.

20. Гунько, Катерина Дмитрівна. "Усталені порівняння в сучасних українській і англійській мовах: структурно-семантичний і лінгвокультурологічний аналіз." (2023).

21. Node.js Documentation URL : https://nodejs.org/en/docs (дата звернення 24.11.2023)

22. «MySQL Documenatation» URL: https://dev.mysql.com/doc/ (дата звернення 20.11.2023)

23. phpMyAdmin URL : https://www.phpmyadmin.net/ (дата звернення 19.11.2023)

24. «DotEnv Documentation» URL: https://www.dotenv.org/docs/ (дата звернення 19.11.2023)

25. Axios Documentation URL: https://axios-http.com/docs/intro (дата звернення 01.12.2023)

26. Cheerio Documentation URL: https://cheerio.js.org/ (дата звернення 01.12.2023)

27. Natural Documentation URL: https://www.npmjs.com/package/natural (дата звернення 01.12.2023)

28. Волобуєва, Владлена Віталіївна. "Дослідження методів приведення слів до нормальної форми для проведення семантичного аналізу тексту." (2020).

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

ДОДАТКИ

Додаток А

ПРОТОКОЛ ПЕРЕВІРКИ
МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Система пошуку та аналізу небезпечного контенту інформаційних ресурсів

Автор роботи: П`ятак Руслан Олегович

Тип роботи: магістерська кваліфікаційна робота

Підрозділ кафедра захисту інформації ФІТКІ
(кафедра, факультет)

Показники звіту подібності Unicheck

Оригінальність – 80,8 %. Схожість – 19,2 %.

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 _____ Валентина КАПЛУН
(підпис)

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

Автор роботи _____ Руслан П’ЯТАК
(підпис)

Керівник роботи _____ Олеся ВОЙТОВИЧ
(підпис)

ДОДАТОК Б

Текст програми серверної частини

```
<!DOCTYPE html>

<head>
  <meta charset="UTF-8">
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      width: 500px;
      height: 600px;
    }

    .popup {
      width: 100vw;
      height: 100vh;
      background-color: #a1261d;
      padding: 20px;
      border-radius: 10px;
      position: fixed;
      top: 50%;
      left: 50%;
      transform: translate(-50%, -50%);
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
      text-align: center;
    }

    h1 {
      color: #fff;
    }

    .button {
      display: inline-block;
      padding: 10px 20px;
      background-color: transparent;
      color: #fff;
      border: none;
      border-radius: 5px;
      cursor: pointer;
      transition: background-color 0.3s ease-in-out;
    }

    .button:hover {
      transform: scale(1.05);
    }
```

```
    p {
      color: #fff;
    }

    ul {
      list-style: none;
      padding: 0;
    }

    li {
      color: #fff;
      margin: 5px 0;
    }

    .enabled {
      color: #a1261d;
    }
  </style>
</head>

<body>

  <div class="popup" id="popup">
    <h1>BadContentBlocker</h1>
    <button class="button" id="toggleButton"><img id="image" src="/images/icon128.png" alt=""></button>
    <p style="font-size: 24px;" id="status">ЗАХИСТ ВИМКНЕНО</p>
    <hr>
    <p style="font-size: 24px;">Найбільш небезпечні сайти:</p>
    <ul id="siteList"></ul>
    <button class="button" id="allSites">Всі перевірені сайти</button>
    <hr>
    <p style="font-size: 24px;">Перевірити конкретне джерело:</p>
    <input type="text" id="siteInput">
    <button class="button" id="findButton">Перевірити</button>
    <script src="Popup.js" nonce="Script"></script>
  </div>

</body>

</html>

var protectionEnabled = false;

document.getElementById('toggleButton').addEventListener('click', function () {
  var popup = document.getElementById('popup');
  var button = document.getElementById('image')
  var status = document.getElementById('status');

  protectionEnabled = !protectionEnabled;
```

```
  if (protectionEnabled) {
    popup.style.backgroundColor = '#33cc99';
    button.src = '/images/checked128.png'
    status.innerHTML = 'ЗАХИСТ ВВІМКНЕНО';
  } else {
    popup.style.backgroundColor = '#a1261d';
    button.src = '/images/icon128.png'
    status.innerHTML = 'ЗАХИСТ ВИМКНЕНО';
  }
})

document.getElementById('findButton').addEventListener('click', function () {
  var input = document.getElementById('siteInput');

  var isConfirmed = confirm(`Ви впевнені що хочете перевірити сайт: ${input.value}`)

  if (isConfirmed) {
    setTimeout(() => {
      alert('Виявлено мову ворожнечі! Рівень 5, покиньте сторінку або ввімкніть онлайн захист.')
    }, 3000)
  }
})

const download = function (data) {
  const blob = new Blob([data], { type: 'text/csv' });
  const url = window.URL.createObjectURL(blob)
  const a = document.createElement('a')
  a.setAttribute('href', url)
  a.setAttribute('download', 'download.csv');
  a.click()
}

const csvmaker = function (data) {
  csvRows = [];

  const headers = Object.keys(data[0])

  csvRows.push(headers.join(','));

  for (var i = 0; i < data.length; i++) {
    const values = Object.values(data[i]).join(',')
    csvRows.push(values)
  }

  return csvRows.join('\n')
}

const get = async function () {
```

```
  const data = [{
    id: 'test',
    name: 'test',
    profsdas: 'test'
  }, {
    id: 'test1',
    name: 'test1',
    profsdas: 'test1'
  }, {
    id: 'test1',
    name: 'test1',
    profsdas: 'test1'
  }, {
    id: 'test1',
    name: 'test1',
    profsdas: 'test1'
  }, {
    id: 'test1',
    name: 'test1',
    profsdas: 'test1'
  }, {
    id: 'test1',
    name: 'test1',
    profsdas: 'test1'
  }, {
    id: 'test1',
    name: 'test1',
    profsdas: 'test1'
  }]

  const csvdata = csvmaker(data);
  download(csvdata);
}

document.getElementById('allSites').addEventListener('click', async () => {
  const isConfirmed = confirm('Ви впевнені, що хочете завантажити csv файл зі звітом всіх перевірених сайтів?')

  if (isConfirmed) {
    await get();
  }
})

var siteList = document.getElementById('siteList');
var sites = ['omegle.com', 'hdrezka.ru'];

sites.forEach(function (site) {
  var li = document.createElement('li');
  li.textContent = site;
  siteList.appendChild(li);
});
```

```
@import url(https://fonts.googleapis.com/css?family=opensans:500);

body {
  background: #33cc99;
  color: #fff;
  font-family: "Open Sans", sans-serif;
  max-height: 700px;
  overflow: hidden;
}

.c {
  text-align: center;
  display: block;
  position: relative;
  width: 80%;
  margin: 100px auto;
}

._404 {
  font-size: 220px;
  position: relative;
  display: inline-block;
  z-index: 2;
  height: 250px;
  letter-spacing: 15px;
}

._1 {
  text-align: center;
  display: block;
  position: relative;
  letter-spacing: 12px;
  font-size: 4em;
  line-height: 80%;
}

._2 {
  text-align: center;
  display: block;
  position: relative;
  font-size: 20px;
}

.text {
  font-size: 70px;
  text-align: center;
  position: relative;
  display: inline-block;
  margin: 19px 0px 0px 0px;
```

```
  /* top: 256.301px; */
  z-index: 3;
  width: 100%;
  line-height: 1.2em;
  display: inline-block;
}

.btn {
  background-color: rgb(255, 255, 255);
  position: relative;
  display: inline-block;
  width: 358px;
  padding: 5px;
  z-index: 5;
  font-size: 25px;
  margin: 0 auto;
  color: #33cc99;
  text-decoration: none;
  margin-right: 10px;
}

.right {
  float: right;
  width: 60%;
}

hr {
  padding: 0;
  border: none;
  border-top: 5px solid #fff;
  color: #fff;
  text-align: center;
  margin: 0px auto;
  width: 420px;
  height: 10px;
  z-index: -10;
}

hr:after {
  content: "\2022";
  display: inline-block;
  position: relative;
  top: -0.75em;
  font-size: 2em;
  padding: 0 0.2em;
  background: #33cc99;
}

.cloud {
  width: 350px;
```

```
  height: 120px;

  background: #fff;
  background: linear-gradient(top, #fff 100%);
  background: -webkit-linear-gradient(top, #fff 100%);
  background: -moz-linear-gradient(top, #fff 100%);
  background: -ms-linear-gradient(top, #fff 100%);
  background: -o-linear-gradient(top, #fff 100%);

  border-radius: 100px;
  -webkit-border-radius: 100px;
  -moz-border-radius: 100px;

  position: absolute;
  margin: 120px auto 20px;
  z-index: -1;
  transition: ease 1s;
}

.cloud:after,
.cloud:before {
  content: "";
  position: absolute;
  background: #fff;
  z-index: -1;
}

.cloud:after {
  width: 100px;
  height: 100px;
  top: -50px;
  left: 50px;

  border-radius: 100px;
  -webkit-border-radius: 100px;
  -moz-border-radius: 100px;
}

.cloud:before {
  width: 180px;
  height: 180px;
  top: -90px;
  right: 50px;

  border-radius: 200px;
  -webkit-border-radius: 200px;
  -moz-border-radius: 200px;
}

.x1 {
```

```
  top: -50px;
  left: 100px;
  -webkit-transform: scale(0.3);
  -moz-transform: scale(0.3);
  transform: scale(0.3);
  opacity: 0.9;
  -webkit-animation: moveclouds 15s linear infinite;
  -moz-animation: moveclouds 15s linear infinite;
  -o-animation: moveclouds 15s linear infinite;
}

.x1_5 {
  top: -80px;
  left: 250px;
  -webkit-transform: scale(0.3);
  -moz-transform: scale(0.3);
  transform: scale(0.3);
  -webkit-animation: moveclouds 17s linear infinite;
  -moz-animation: moveclouds 17s linear infinite;
  -o-animation: moveclouds 17s linear infinite;
}

.x2 {
  left: 250px;
  top: 30px;
  -webkit-transform: scale(0.6);
  -moz-transform: scale(0.6);
  transform: scale(0.6);
  opacity: 0.6;
  -webkit-animation: moveclouds 25s linear infinite;
  -moz-animation: moveclouds 25s linear infinite;
  -o-animation: moveclouds 25s linear infinite;
}

.x3 {
  left: 250px;
  bottom: -70px;

  -webkit-transform: scale(0.6);
  -moz-transform: scale(0.6);
  transform: scale(0.6);
  opacity: 0.8;

  -webkit-animation: moveclouds 25s linear infinite;
  -moz-animation: moveclouds 25s linear infinite;
  -o-animation: moveclouds 25s linear infinite;
}

.x4 {
  left: 470px;
```

```
 botttom: 20px;

 -webkit-transform: scale(0.75);
 -moz-transform: scale(0.75);
 transform: scale(0.75);
 opacity: 0.75;

 -webkit-animation: moveclouds 18s linear infinite;
 -moz-animation: moveclouds 18s linear infinite;
 -o-animation: moveclouds 18s linear infinite;
}

.x5 {
 left: 200px;
 top: 300px;

 -webkit-transform: scale(0.5);
 -moz-transform: scale(0.5);
 transform: scale(0.5);
 opacity: 0.8;

 -webkit-animation: moveclouds 20s linear infinite;
 -moz-animation: moveclouds 20s linear infinite;
 -o-animation: moveclouds 20s linear infinite;
}

@-webkit-keyframes moveclouds {
 0% {
  margin-left: 1000px;
 }

 100% {
  margin-left: -1000px;
 }
}

@-moz-keyframes moveclouds {
 0% {
  margin-left: 1000px;
 }

 100% {
  margin-left: -1000px;
 }
}

@-o-keyframes moveclouds {
 0% {
  margin-left: 1000px;
 }
```

```
  100% {
    margin-left: -1000px;
  }
}

import Tesseract from 'tesseract.js';


function recognize(file, lang, logger) {
  return Tesseract.recognize(file, lang, {logger})
    .then(({ data: {text }}) => {
      return text;
    })
}

const log = document.getElementById('log');

function updateProgress(data) {
  log.innerHTML = '';
  const statusText = document.createTextNode(data.status);
  const progress = document.createElement('progress');
  progress.max = 1;
  progress.value = data.progress;
  log.appendChild(statusText);
  log.appendChild(progress);
}


function setResult(text) {
  log.innerHTML = '';
  text = text.replace(/\n\s*\n/g, '\n');
  const pre = document.createElement('pre');
  pre.innerHTML = text;
  log.appendChild(pre);
}

document.getElementById('start').addEventListener('click', () => {
  const file = document.getElementById('file').files[0];
  if (!file) return;

  const lang = document.getElementById('langs').value;

  recognize(file, lang, updateProgress)
    .then(setResult);
});

var natural = require('natural');
var tokenizer = new natural.WordTokenizer();
```

```
console.log(tokenizer.tokenize("The quick brown fox jumps over the lazy dog"));


var natural = require('natural');
var Analyzer = natural.SentimentAnalyzer;
var stemmer = natural.PorterStemmer;
var analyzer = new Analyzer("Ukrainian", stemmer, "afinn");

const generateSTYLES = () => {
 return `<style>@import url(https://fonts.googleapis.com/css?family=opensans:500);
 body {
  background: #a1261d;
  color: #fff;
  font-family: "Open Sans", sans-serif;
  max-height: 700px;
  overflow: hidden;
 }
 .c {
  text-align: center;
  display: block;
  position: relative;
  width: 80%;
  margin: 100px auto;
 }
 ._404 {
  font-size: 220px;
  position: relative;
  display: inline-block;
  z-index: 2;
  height: 250px;
  letter-spacing: 15px;
 }
 ._1 {
  text-align: center;
  display: block;
  position: relative;
  letter-spacing: 12px;
  font-size: 4em;
  line-height: 80%;
 }
 ._2 {
  text-align: center;
  display: block;
  position: relative;
  font-size: 20px;
 }
 .text {
  font-size: 70px;
  text-align: center;
  position: relative;
```

```
  display: inline-block;
  margin: 19px 0px 0px 0px;
  /* top: 256.301px; */
  z-index: 3;
  width: 100%;
  line-height: 1.2em;
  display: inline-block;
}


.right {
  float: right;
  width: 60%;
}

hr {
  padding: 0;
  border: none;
  border-top: 5px solid #fff;
  color: #fff;
  text-align: center;
  margin: 0px auto;
  width: 420px;
  height: 10px;
  z-index: -10;
}

hr:after {
  display: inline-block;
  position: relative;
  top: -0.75em;
  font-size: 2em;
  padding: 0 0.2em;
  background: #33cc99;
}

.cloud {
  width: 350px;
  height: 120px;

  background: #a1261d; /* Ý¡Ý²Ñ–Ñ‚Ý»Ý¾ Ñ‡ÝµÑ€Ý²Ý¾Ý½Ý¸Ý¹ Ý°Ý¾Ý»Ñ–Ñ€ */
  background: linear-gradient(top, #a1261d 100%);
  background: -webkit-linear-gradient(top, #a1261d 100%);
  background: -moz-linear-gradient(top, #a1261d 100%);
  background: -ms-linear-gradient(top, #a1261d 100%);
  background: -o-linear-gradient(top, #a1261d 100%);

  border-radius: 100px;
  -webkit-border-radius: 100px;
  -moz-border-radius: 100px;
```

```
  position: absolute;
  margin: 120px auto 20px;
  z-index: -1;
  transition: ease 1s;
}

.cloud:after,
.cloud:before {
  content: "";
  position: absolute;
  background: #a1261d;
  z-index: -1;
}

.cloud:after {
  width: 100px;
  height: 100px;
  top: -50px;
  left: 50px;

  border-radius: 100px;
  -webkit-border-radius: 100px;
  -moz-border-radius: 100px;
}

.cloud:before {
  width: 180px;
  height: 180px;
  top: -90px;
  right: 50px;

  border-radius: 200px;
  -webkit-border-radius: 200px;
  -moz-border-radius: 200px;
}

.x1 {
  top: -50px;
  left: 100px;
  -webkit-transform: scale(0.3);
  -moz-transform: scale(0.3);
  transform: scale(0.3);
  opacity: 0.9;
  -webkit-animation: moveclouds 15s linear infinite;
  -moz-animation: moveclouds 15s linear infinite;
  -o-animation: moveclouds 15s linear infinite;
}

.x1_5 {
```

```
  top: -80px;
  left: 250px;
  -webkit-transform: scale(0.3);
  -moz-transform: scale(0.3);
  transform: scale(0.3);
  -webkit-animation: moveclouds 17s linear infinite;
  -moz-animation: moveclouds 17s linear infinite;
  -o-animation: moveclouds 17s linear infinite;
}

.x2 {
  left: 250px;
  top: 30px;
  -webkit-transform: scale(0.6);
  -moz-transform: scale(0.6);
  transform: scale(0.6);
  opacity: 0.6;
  -webkit-animation: moveclouds 25s linear infinite;
  -moz-animation: moveclouds 25s linear infinite;
  -o-animation: moveclouds 25s linear infinite;
}

.x3 {
  left: 250px;
  bottom: -70px;

  -webkit-transform: scale(0.6);
  -moz-transform: scale(0.6);
  transform: scale(0.6);
  opacity: 0.8;

  -webkit-animation: moveclouds 25s linear infinite;
  -moz-animation: moveclouds 25s linear infinite;
  -o-animation: moveclouds 25s linear infinite;
}

.x4 {
  left: 470px;
  botttom: 20px;

  -webkit-transform: scale(0.75);
  -moz-transform: scale(0.75);
  transform: scale(0.75);
  opacity: 0.75;

  -webkit-animation: moveclouds 18s linear infinite;
  -moz-animation: moveclouds 18s linear infinite;
  -o-animation: moveclouds 18s linear infinite;
}
```

```
.x5 {
  left: 200px;
  top: 300px;

  -webkit-transform:  scale(0.5);
  -moz-transform: scale(0.5);
  transform: scale(0.5);
  opacity: 0.8;

  -webkit-animation: moveclouds 20s linear infinite;
  -moz-animation: moveclouds 20s linear infinite;
  -o-animation: moveclouds 20s linear infinite;
 }

 @-webkit-keyframes moveclouds {
  0% {
    margin-left: 1000px;
  }
  100% {
    margin-left: -1000px;
  }
 }
 @-moz-keyframes moveclouds {
  0% {
    margin-left: 1000px;
  }
  100% {
    margin-left: -1000px;
  }
 }
 @-o-keyframes moveclouds {
  0% {
    margin-left: 1000px;
  }
  100% {
    margin-left: -1000px;
  }
 }
  </style>`;
};

const generateHTML = (pageName) => {
 return `

 <div class='c'>
    <div class='_404'>Загроза</div>
    <hr>
    <div class='_1'>Виявлено небезпечний контент</div>
 </div>
  `;};
```

Додаток В

ІЛЮСТРАТИВНА ЧАСТИНА

СИСТЕМА ПОШУКУ ТА АНАЛІЗУ НЕБЕЗПЕЧНОГО КОНТЕНТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

Узагальнена архітектура системи

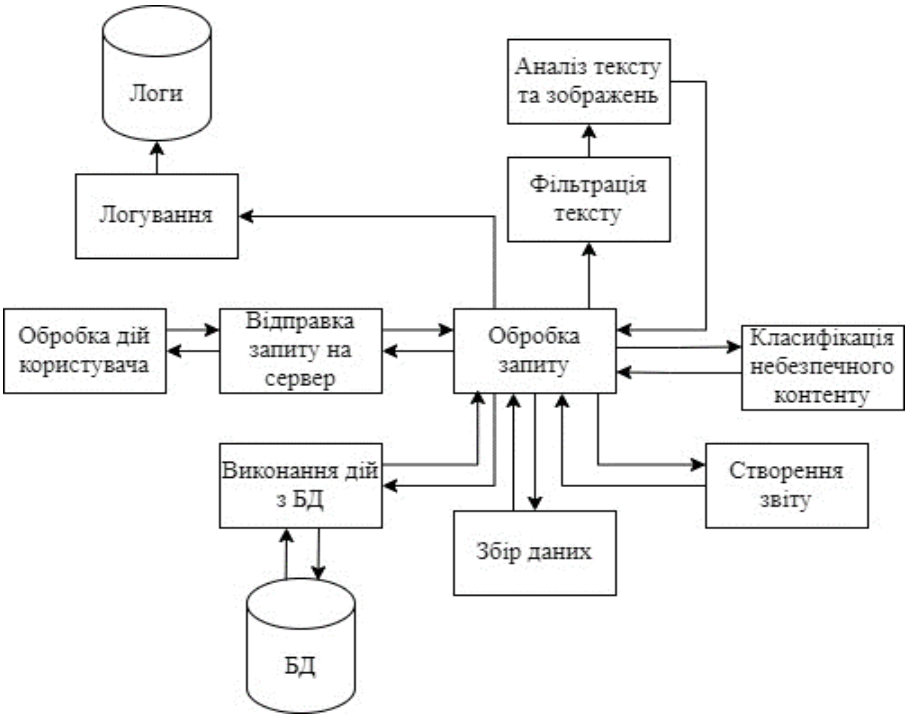


Схема роботи модулю взаємодії з користувачам

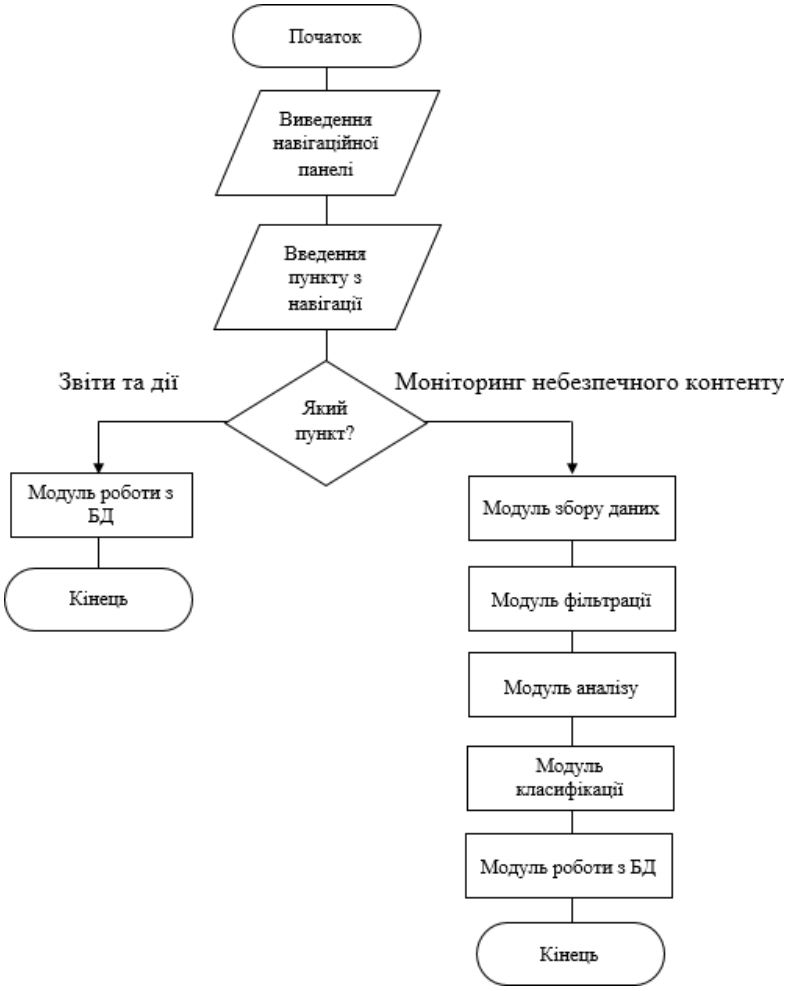


Схема роботи модуля збору даних

Схема роботи модулю запису в базу даних

Схема роботи модулю фільтрації

Схема роботи модулю аналізу тексту та зображень

Схема роботи модулю класифікації

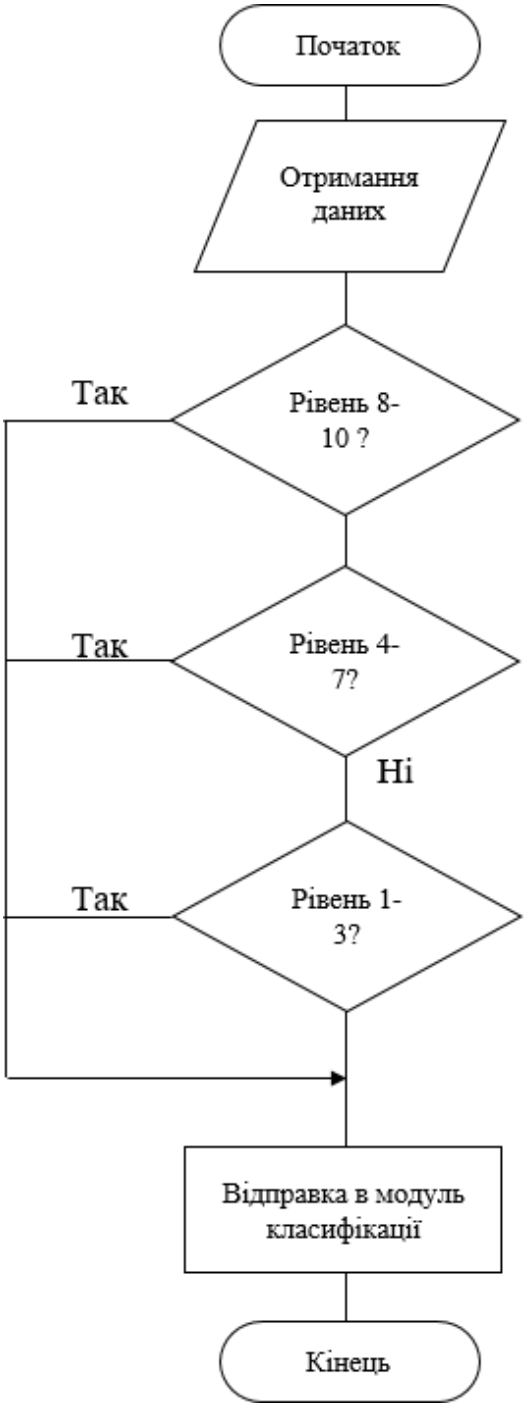
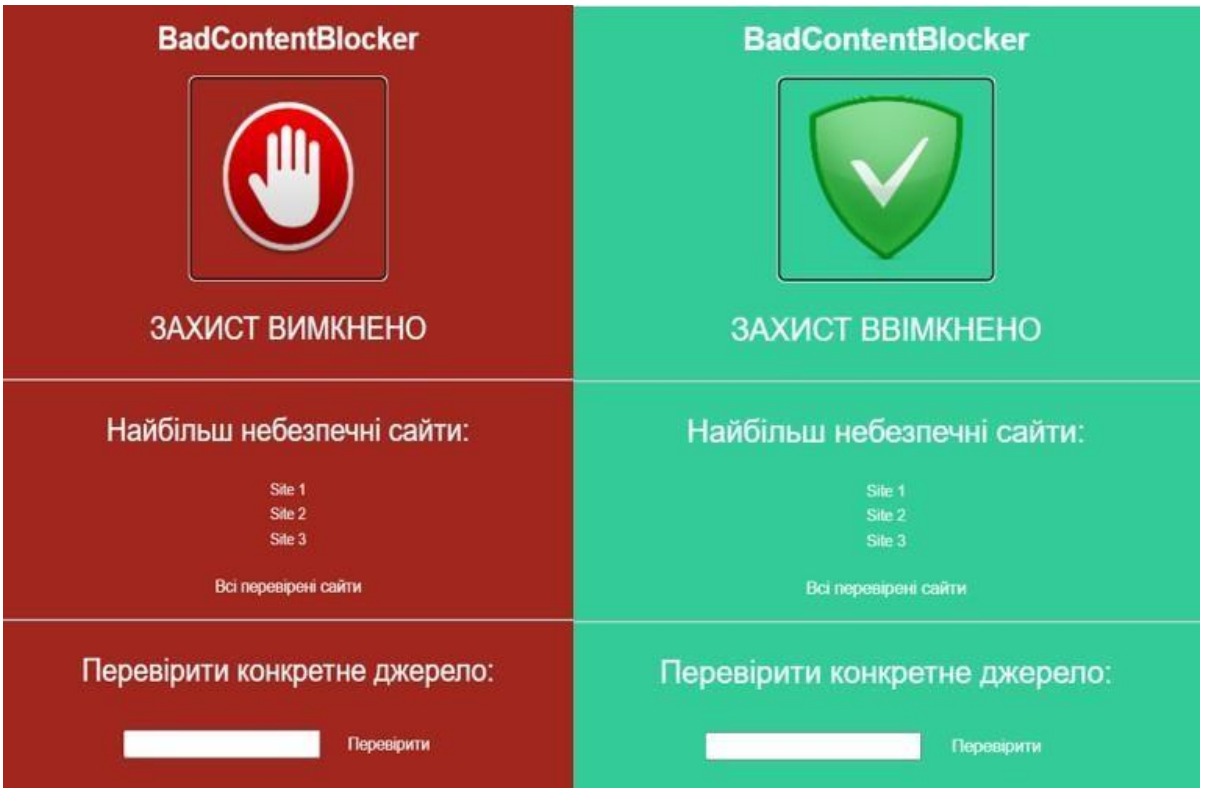


Схема роботи модулю звітування

Вигляд веб-розширення одразу після завантаження



Робота режиму онлайн захисту