

Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет  
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”

Завідувач кафедри кібербезпеки  
та програмного забезпечення  
д.т.н., професор

Олексій СМІРНОВ

“ \_\_\_\_ ” \_\_\_\_\_ 2021 р.

**ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА**  
**за другим (магістерським) рівнем вищої освіти**  
на тему

**“Дослідження та програмна реалізація системи кібербезпеки  
для виявлення вразливостей web-застосунків”**

Виконав здобувач вищої освіти  
II курсу, групи КІ-20М-1,4  
ОПП «Комп’ютерна інженерія»  
спеціальності 123 «Комп’ютерна інженерія»  
Толмачов Ю.П.

« \_\_\_\_ » \_\_\_\_\_ 2021 р.

Керівник проекту  
кандидат технічних наук, доцент  
Олександр КОВАЛЕНКО

« \_\_\_\_ » \_\_\_\_\_ 2021 р.

Рецензент \_\_\_\_\_  
\_\_\_\_\_

Центральноукраїнський національний технічний університет  
Факультет Механіко-технологічний  
Кафедра Кібербезпеки та програмного забезпечення  
Рівень вищої освіти магістр  
Галузь знань . 12 “Інформаційні технології”  
Спеціальність 123 “Комп’ютерна інженерія”  
Освітньо-професійна (освітньо-наукова) програма “Комп’ютерна інженерія”

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
д.т.н., проф.

Олексій СМІРНОВ  
« 6 » вересня 2021 року

## ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Толмачову Юрію Петровичу

(прізвище, ім’я, по батькові)

1. Тема роботи Дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків

2. Керівник роботи Коваленко Олександр Володимирович, докт. техн. наук, доцент  
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 42-13 від 02.08.2021 року

3. Строк подання студентом роботи до захисту 10.12.2021 р.

4. Мета та завдання випускної кваліфікаційної роботи: Метою розробки є дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

- |  |   |
|--|---|
| <u>1. Призначення та область використання.</u>           | <u>7. Економічна ефективність розробленої програми.</u> |
| <u>2. Перегляд аналогічних існуючих систем.</u>          | <u>8. Заходи з охорони праці та техніки безпеки</u>     |
| <u>3. Опис і обґрунтування проектних рішень.</u>         | <u>9. Висновки.</u>                                     |
| <u>4. Етапи програмування системи.</u>                   |   |
| <u>5. Впровадження системи в промислову експлуатацію</u> |   |

6. Наукова новизна

6. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

Наукова новизна 1 аркуш

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

Показники економічної ефективності 1 аркуш

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Савеленко Г.В.	05.10.2021	14.11.2021
Охорона праці	Оришака О.В.	06.10.2021	16.11.2021

7. Дата видачі завдання « 6 » вересня 2021 р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2021 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2021 р.	
3.	Розробка моделі компонента	20.10.2021 р.	
4.	Розробка структур даних	25.10.2021 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2021 р.	
6.	Програмування алгоритмів	10.11.2021 р.	
7.	Розрахунок економічної ефективності	13.11.2021 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2021 р.	
9.	Оформлення ПЗ	17.11.2021 р.	
10.	Попередній захист роботи	10.12.2021 р.	

Дата видачі завдання  
« 6 » вересня 2021 р.

Підпис керівника

\_\_\_\_\_ (прізвище та ініціали)

Завдання прийнято до виконання  
« 6 » вересня 2021 р.

Підпис здобувача

\_\_\_\_\_ (прізвище та ініціали)

## АНОТАЦІЯ

**Толмачов Ю.П. Дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2021.**

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи кібербезпеки для виявлення вразливостей web-застосунків.

Метою розробки є дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

Об'єктом дослідження є процес кібербезпеки для виявлення вразливостей web-застосунків.

Предметом дослідження є методи кібербезпеки для виявлення вразливостей web-застосунків.

Методи дослідження базуються на методах теорії захисту інформації, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows XP/Vista/7/8/10.

Програму розроблено в середовищі Java.

**Ключові слова:** комп'ютерна інженерія, вразливості, web-застосунки

## ABSTRACT

**Tolmachov Yu.P. Research and software implementation of a cybersecurity system to detect vulnerabilities in web applications. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2021**

In this final qualification work for the second (master's) level of higher education, software has been developed that is designed for the cybersecurity system to detect vulnerabilities in web applications.

The purpose of the development is to research and software implementation of a cybersecurity system to identify vulnerabilities in web applications.

The object of research is the process of cybersecurity to detect vulnerabilities in web applications.

The subject of the study is cybersecurity methods to detect vulnerabilities in web applications.

Research methods are based on methods of information security theory, methods of mathematical statistics, methods of software development.

The result is a software implementation of a cybersecurity system to detect vulnerabilities in web applications.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

Developed user-friendly interface. Instructions for working with software are given.

The program can be used on an IBM PC with Windows XP / Vista / 7/8/10.

The program is developed in the Java environment.

**Keywords:** computer engineering, vulnerabilities, web applications

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ .....	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ.....	7
1.1 Призначення системи.....	7
1.2 Область застосування.....	8
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ .....	9
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти .....	9
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	16
2.3 Розгорнута постановка завдання .....	21
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ .....	23
3.1 Опис функціонування системи.....	23
3.2 Розробка структурної схеми .....	27
3.3 Розробка функціональної схеми.....	28
3.4 Розробка діаграми процесів .....	29
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ ....	30
4.1 Розробка блок-схем та опис алгоритмів функціонування системи .....	30
4.2 Захист розробленого програмного забезпечення .....	38
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ.....	41
6 НАУКОВА НОВИЗНА .....	43

**ВКРМ-123.21.0022.00.00.ПЗ**

Вим.	Арк.	№ докум.	Підп.	Дата				
Розроб.		Толмачов Ю.П.			Дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків	Лім.	Аркуш	Аркушів
Перев.		Коваленко О.В.				М	1	86
Н.контр.		Гермак В.С.			ЦНТУ КІ-20М-1,4			
Затв.		Смірнов О.А.						

7 ЕКОНОМІЧНА ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ.....	44
7.1 Техніко економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. ....	44
7.2 Розрахунок трудомісткості розробки програмної продукції .....	46
7.3 Визначення чисельності виконавців і планового фонду зарплати .....	48
7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника .....	53
7.5 Визначення собівартості розробки та ціни програмної продукції. ....	57
7.6 Визначення об'єму капітальних вкладень та експлуатаційних витрат у споживача програмної продукції.....	60
7.7 Визначення експлуатаційних витрат.....	61
7.8 Визначення економічної ефективності програмної продукції.....	62
7.9 Висновок. ....	64
8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ .....	65
8.1 Вступ.....	65
8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером .....	66
8.3 Аналіз санітарно-гігієнічних умов праці на робочому місці програміста ..	67
8.4 Розробка заходів з умов поліпшення охорони праці.....	70
8.5 Розрахункова частина .....	71
8.6 Висновки до розділу.....	73
9 ОСНОВНІ ВИСНОВКИ.....	74
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	76

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ІС – інформаційні системи

ЗІ – захист інформації

ІБ – інформаційна безпека

НСД – несанкціонований доступ

ІП – інформаційне повідомлення

ПК – персональний комп'ютер

ОС – операційна система

QA – (quality assurance) забезпечення якості

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

## ВСТУП

**Актуальність теми.** Web-застосунки в сучасному світі знайшли своє місце практично у всіх сферах діяльності людини. Вони використовуються для найрізноманітніших цілей від навчання та розваг до проектування найскладніших систем.

Проте є і зворотна сторона такої широкої функціональності: їх компрометація може призвести до катастрофічних наслідків. Все нові й нові вектори атак ставлять під загрозу безпеку як власника бізнесу так і їх клієнтів.

Для звичайного користувача це може обернутись крадіжкою особистих даних. Щодо компаній – то це може призвести до того, що вона втратить репутацію, позбудеться важливих клієнтів, зазнає фінансових втрат. Несанкціонований доступ до веб-сайту зловмисником загрожує власнику блокуванням сайту в пошукових і рекламних мережах, відмовою в наданні послуг хостингової компанії, втратою клієнтської бази і зруйнованої діловою репутацією. Відвідувачі веб-сайту ризикують потрапити під контроль "хакерів" в разі розміщення ними шкідливого коду на його сторінках. Зловмисники можуть отримати ваші приватні дані, зашифрувати всі файли на пристрої або ж проводити стеження за користувачами.

Постійно зростаючий функціонал, проведення платежів, передача персональних даних та інше, все це робить сайти все більше привабливими для різного роду порушників та крадіїв інформації, що обумовлює необхідність розробки і впровадження нових та вдосконалення існуючих засобів захисту.

Саме тому, виявлення вразливостей безпеки web-застосунку не менш важлива, ніж реалізація його основних функцій. Виявлення вразливостей на етапі проектування та розробки web-застосунку дозволяє значно полегшити роботу над його системою захисту.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

**Мета й завдання дослідження.** Метою роботи є дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем кібербезпеки для виявлення вразливостей web-застосунків.
- Дослідження системи кібербезпеки для виявлення вразливостей web-застосунків.
- Програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

*Об'єктом дослідження* є процес кібербезпеки для виявлення вразливостей web-застосунків.

*Предметом дослідження* є методи кібербезпеки для виявлення вразливостей web-застосунків.

*Методи дослідження* базуються на методах теорії захисту інформації, методах математичної статистики, методах розробки програмного забезпечення.

**Наукова новизна отриманих результатів.** У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод кібербезпеки для виявлення вразливостей web-застосунків.
- Розроблено вітчизняний продукт кібербезпеки для виявлення вразливостей web-застосунків, який має більш широкі можливості, на відміну від існуючих аналогів.

**Практична цінність отриманих результатів** полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі кібербезпеки для виявлення вразливостей web-застосунків.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

**Достовірність наукових результатів** підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LV Науково-технічна конференція здобувачів вищої освіти «Наука – виробництву», 2021, основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №12.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

# 1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

## 1.1 Призначення системи

Веб-застосунки вже давно є улюбленими мішенями зловмисників. Вони можуть забезпечити доступ до коштовної інформації. Крім того, їх порівняно легко використовувати. Успішна атака може мати катастрофічні наслідки, включаючи фінансові збитки, втрату репутації бренду й довіри клієнтів. Деякі організації вже не можуть відновитися після серйозного порушення безпеки.

Міжмережевий екран виконує роль першої лінії захисту центру обробки даних, однак не слід обмежуватися тільки цим застосунком. У зв'язку з появою нових атак, орієнтованих на поразку застосунків і кінцевих користувачів, виникає необхідність у застосункових функціях захисту, яких позбавлені міжмережеві екрани й застосунки IPS. Методи виявлення, засновані на підписах, служби репутації IP-адрес і функції поглибленої перевірки пакетів забезпечують захист від деяких просунутих загроз, однак вони недостатньо широко представлені на ринку. Для захисту корпоративних мереж потрібні застосункові продукти, такі як міжмережеві екрани веб-застосунків, контролери доставки застосунків і інтегровані «пісочниці», що забезпечують протидію новим загрозам, цілями яких є центри обробки даних і користувачі.

У порівнянні з автономними застосунками адаптивна система мережної безпеки забезпечує більш ефективний захист веб-застосунків. Наприклад, інтегрована «пісочниця» підвищує продуктивність функції антивірусного сканування, призначеної для виявлення просунутих загроз. Спільне розгортання цього застосунку спрощує впровадження й забезпечує обмін даними про загрози, а також інтеграцію із продуктами провідних постачальників служб сканування наявності загроз, що сприяє підвищенню ефективності виправлення уразливостей.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

## 1.2 Область застосування

Ключові компоненти застосунку:

- Міжмережеві екрани веб-застосунків для виправлення уразливостей веб-застосунків
- Контролери доставки застосунків з високою пропускнуою здатністю для захисту трафіка веб-застосунків
- Засіб усунення наслідків DDoS-атак і захисту від атак, що вражають служби застосунків на рівні 7

Засіб безпеки веб-застосунків відрізняється високими показниками ефективності, продуктивності й інтегрованості, завдяки чому воно ідеально підходить для захисту важливих веб-застосунків від атак, що використовують відомі й невідомі уразливості.

Отже програмне забезпечення магістерської роботи призначене для захисту даних від несанкціонованого доступу при роботі з різного роду веб-застосунками та системами – в навчальних закладах, на підприємствах, в офісах, банківських мережах та інше.

Приведені дослідження та аналіз літературних джерел прискорить засвоєння та закріплення навчального матеріалу з вказаної тематики. Крім того надасть можливість розроблювати більш оптимальні алгоритми програм при розробці власних програмних систем відповідного призначення.

Використані в ході роботи над проектом алгоритми та прийоми програмування, а також супроводжений докладними коментарями програмний код і пояснювальна записка будуть корисні починаючим програмістам і спеціалістам в сфері QA і можуть бути використані при підготовці студентів та фахівців з напрямку кібербезпека.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей веб-застосунків, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

## 2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

### 2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Одним з найбільш відомих продуктів в сфері захисту є Open Web Application Security Project (OWASP) – це відкритий проект забезпечення безпеки web-застосунків.

OWASP не афілійовані з жодною компанією, що займається розробкою технологій, але він підтримує грамотне використання технологій безпеки. Засновники проекту вважають, що свобода від впливу з боку інших організацій може полегшити поширення неупередженої, корисної і дешевої інформації про безпеку web-застосунків.

Учасники спільноти OWASP роблять web-застосунки безпечніше, враховуючи людський фактор і технологічний рівень.

Найбільш затребувані документи, опубліковані OWASP, включають в себе: OWASP Guide Project – OWASP, Category:OWASP Code Review Project і [HTTP://www.owasp.org/index.php/OWASP\\_Top\\_Ten\\_Project](http://www.owasp.org/index.php/OWASP_Top_Ten_Project).

Найпоширенішими інструментами OWASP є тренувальне середовище [HTTP://www.owasp.org/index.php/OWASP\\_WebGoat\\_Project](http://www.owasp.org/index.php/OWASP_WebGoat_Project) WebGoat, проксі-аналізатор WebScarab і .NET інструменти.

OWASP створює стандарти, перший з яких був опублікований під назвою OWASP Application Security Verification Standard (ASVS)). Основна мета OWASP ASVS – це стандартизація діапазону охоплення і рівня суворості доступних на ринку web-застосунків, що забезпечують безпеку. Метою OWASP ASVS також було створення набору комерційно успішних відкритих стандартів, пристосованих для спеціалізованих веб-технологій. Збірник для Web-застосунків

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

вже був опублікований. Збірник для Web-сервісів в процесі розробки.

### **Деякі з проектів OWASP**

Стандарт Підтвердження Безпеки Web-застосувань OWASP (OWASP Application Security Verification Standard (ASVS)) – Стандарт для проведення перевірок рівня безпеки web-застосувань.

Керівництво по Розробці OWASP дає практичні поради і містить приклади коду на J2EE, ASP.NET і PHP. Серйозно перероблене в 2014 році, Керівництво по Розробці охоплює великий масив питань безпеки для рівня web-застосувань, від SQL ін'єкцій до сучасних проблем, таких як фішинг, обробка кредитних карт, фіксація сесій, підробка міжсайтових запитів, узгодження і конфіденційність.

Керівництво з Тестування OWASP включає «кращу практичну» основу для тестування проникнень, яку користувачі можуть використовувати в своїх організаціях і «низькорівневе» керівництво по тестуванню проникнень, яке описує техніки тестування найбільш поширених проблем з безпекою в web-застосунких і web-сервісах.

OWASP ЗАП Проект: Проксі Зет-атаки – це простий в застосуванні вбудований інструмент тестування проникнень, служить для знаходження вразливостей веб-застосувань. Він розроблений для використання людьми з різним досвідом в сфері безпеки і є еталоном для розробників і тестувальників функціоналу, які не мають досвіду в тестуванні проникнень.

OWASP Топ-10: мета проекту Топ-10 – збільшення обізнаності про безпеку web-застосувань за допомогою визначення найбільш критичних ризиків, що загрожують організаціям. На проект Топ-10 посилається безліч стандартів, інструментів і організацій, включаючи MITRE, PCI DSS, DISA, FTC, і безліч інших.

Webgoat – свідомо ненадійний веб-застосунок, створений OWASP як керівництво з написання безпечного коду. Разом з застосунком поставляється підручник і набір різних курсів, які розповідають студентам як використовувати інформацію про уразливість для написання безпечного коду.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10



```
1 # DDoS attacks via other sites execution tool
2 # DAVOSET v.1.1.3
3 # Tool for conducting of DDoS attacks on the sites via other sites
4 # Copyright (C) MustLive 2010-2013
5 # Last update: 31.08.2013
6 # http://websecurity.com.ua
7 #####
8 # Program summary
9 #####
10
11 DAVOSET - it is console (command line) tool for conducting DDoS attacks on the sites via Abuse of
12 Functionality vulnerabilities at other sites.
13
14 About such attacks you can read in my article "Using of the sites for attacks on other sites"
15 (http://websecurity.com.ua/4322/).
16
17 #####
18 # Usage of the tool
19 #####
20
21 To use another site as a proxy (to access or attack one site by connecting via another site):
22
23 perl davoset.pl u=http://site l=one_server.txt
24
25 To conduct DoS attack via another site:
26
27 perl davoset.pl u=http://site/large\_file l=one_server.txt
28
29 To conduct DDoS attack via a list of zombie-servers:
30
31 perl davoset.pl u=http://site l=many_servers.txt
32
33 #####
34 # Attacking on the site
```

Рисунок 2.2 – Інтерфейс DAVOSET

«DDoS attacks via other sites execution tool (DAVOSET) – це інструмент для використання Abuse of Functionality та XML External Entities уразливостей на одних сайтах для проведення атак на інші сайти (зокрема DoS і DDoS атак)» [2].

Досить цікавий продукт Variti – сервіс, що захищає ресурси від шкідливих ботів без втрати живих клієнтів. Допомогає врятуватися від DDoS -атак, фрода і парсинга, захищає API і проводить аудит безпеки. Власна технологія фільтрації трафіку дозволяє блокувати не IP-адреси цілком, а окремі бот-запити, що робить сайти більш доступними і безпечними для користувачів, при цьому не впливаючи на їх індексацію пошуковими системами.

Послугами Variti користуються компанії зі сфер e-commerce, фінансів, хостингу та медіа. Сервіс працює на власній хмарної інфраструктурі, ніяк не навантажує клієнтські сервери і не вимагає змін в коді продукту.

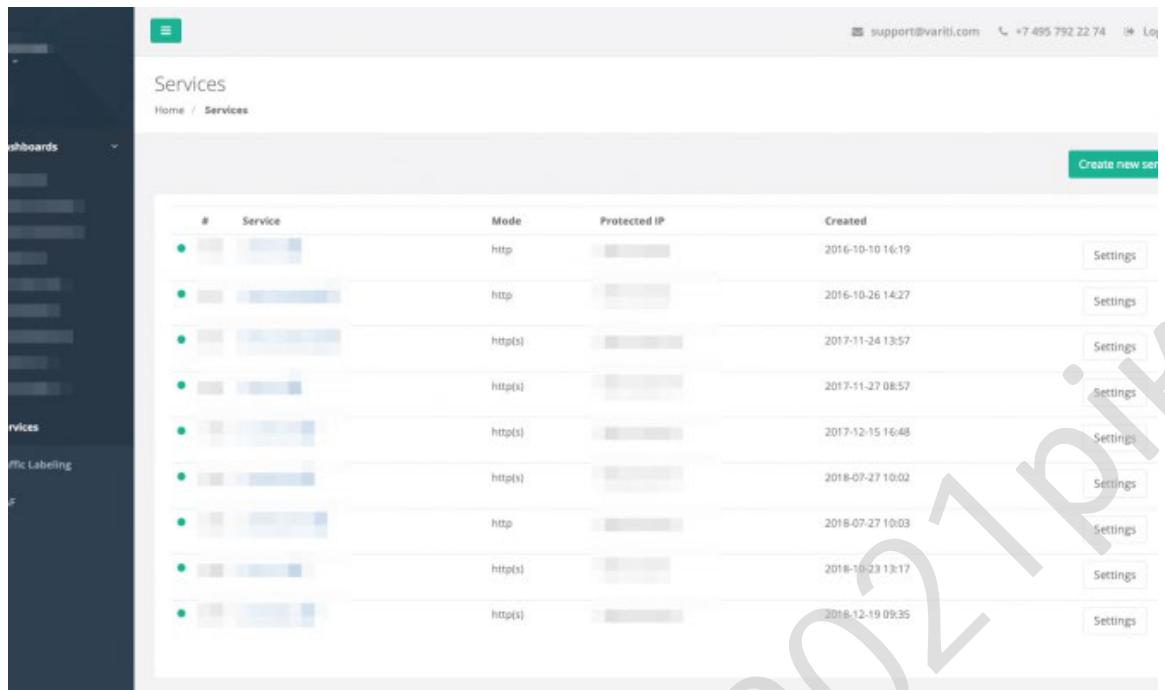


Рисунок 2.3 – Інтерфейс сервісу Variti

Основні функції:

- Захист від DDoS-атак (від високо і низькочастотних атак)
- Захист від ботів (рекламного фрода, перебору паролів, парсинга)
- Захист API (від автореєстрації, злому, крадіжки даних)
- Аудит безпеки (навантажувальний і стрес-тестування, тестування на проникнення)

До основних відмінностей сервісу відносяться:

- Захист від атак на рівні L7 з першого запиту, без блокування IP-адрес
- Відсутність CAPTCHA для визначення ботів
- Захист HTTPS без розкриття SSL і передачі логів
- Захист мобільного API без змін в коді програми
- Підтримка веб-сокетів

**Який продукт вибрати?**

Незважаючи на те, що застосунки Web Application Firewall у світі вже встигнули набрати достатню популярність, і такі компанії як Amazon надають

захищені сервіси на базі продуктів даного класу, на ринку України для багатьох використання WAF ще в новинку. Сильними гравцями на ринку є Fortinet з застосунком Fortiweb і Imperva з Securesphere Web Application Firewall.

Міжмережеві екрани для веб-застосунків Fortiweb від Fortinet орієнтовані на використання в компаніях середнього й великого бізнесу, а також сервісів-провайдерів. У продукті застосовується система двостороннього захисту від впровадження коду (SQL injection) і міжсайтового скриптинга (cross-site scripting) і подібних загроз, завдяки чому Fortiweb дозволяє запобігти крадіжці персональних даних, фінансові махінації й промислове шпигунство. Fortiweb надає інструменти, необхідні для моніторингу й приведення корпоративної політики безпеки у відповідність із вимогами регуляторів. Крім того, продукт суттєво прискорює роботу веб-застосунків за рахунок балансування навантаження на мережу й оптимізації використання мережних ресурсів.

Важливою перевагою Fortiweb є його вартість, яка не залежить від кількості користувачів. Компаніям досить купити один раз пристрій з невеликим запасом по потужності й далі не прийде витрачатися на закупівлю встаткування при збільшенні кількості співробітників.

Також серед гідностей Fortiweb необхідно відзначити той факт, що у своїх апаратних і програмних застосунках компанія Fortinet використовує власні технології. Так, для прискорення обробки й шифрування даних застосовуються спеціалізовані процесори Fortiasic. Крім того, розроблена своя операційна система Fortios, оптимізована під завдання безпеки.

Fortinet якісно супроводжує свою продукцію, постійно виконуючи роботи з виявлення нових загроз у власному аналітичному центрі Fortiguard Security Center. У результаті відновлення сигнатур, чорних списків сайтів і баз репутацій відбувається кілька раз у день.

Важливо також треба відзначити тісну інтеграція всіх пристроїв Fortinet за рахунок повної сумісності один з одним, що дозволяють швидко й просто масштабувати систему. Істотною перевагою систем захисту на базі Fortinet є

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

високий ступінь автоматизації операцій і простота їх супроводу. Це дозволяє скоротити число помилок, викликаних людським фактором і зменшити число співробітників обслуговуючого персоналу.

Imperva Securesphere Web Application Firewall – один з лідерів в області відбиття атак на веб-застосунки. Застосунки від Imperva забезпечують надійний захист за рахунок одночасного застосування декількох технологій у сфері ІБ: сигнатурний аналіз, перевірка протоколів на аномалії, відстеження сесій, динамічне профілювання.

Сімейство продуктів Secure-Sphere WAF успішно протидіють як усім атакам з OWASP-TOP10, так і іншим менш відомим, але більш витонченим атакам. Пристрою мають можливості по інспектуванню зашифрованих даних, що пересилаються по протоколу SSL (HTTPS).

Застосунок складається з наступних модулів:

- Securesphere Web Application Firewall – захист веб-застосунків від кібератак;
- Threatradar – репутаційна база даних.

Secure-Sphere WAF здатний глибоко аналізувати логіку роботи легального веб-застосунки, виконувати інтелектуальне дослідження спроб проникнення й атак; протидіяти HTTP-атакам, включаючи атаки на переповнення буфера, дії зловливих програм і зловмисників. Застосунок оснащений механізмом захисту від хробаків і інших шкідливих атак на веб-сервери й застосунки, основу якого становлять механізми на основі сигнатур популярної системи Snort і власних SQL-сигнатур, розроблювальних дослідним центром ADC (Application Defense Center) компанії Imperva. Вбудований міжмережевий екран здійснює надійний захист від неавторизованих користувачьких запитів і атак на мережному рівні.

Передвстановлені в системі звіти повністю задовольняють вимогам стандартів інформаційної безпеки. Можливе створення користувачьких звітів (у тому числі за розкладом) і експорт у різні формати. Застосункові хмарні сервіси дозволяють спростити безпека й упоратися з DDoS-атаками.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

Важлива перевага пристроїв Securesphere WAF: наявність унікального сервісу Threatradar, що забезпечує захист від автоматизованих атак. Завдяки швидкому одержанню достовірної інформації про джерела атак, Threatradar дозволяє негайно блокувати трафік, що йде від підозрілих джерел, ще до моменту здійснення якої-небудь руйнівної дії. Застосунки Imperva відрізняються прозорою підтримкою й простим розгортанням.

## 2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

В ході розробки програмного забезпечення кваліфікаційної магістерської роботи використано мову програмування Java і середовище розробки IntelliJ IDEA.

Java – суворо типізована об'єктно-орієнтована мова програмування, розроблена компанією Sun Microsystems (в подальшому придбана компанією Oracle). Розробка і супроводження ведеться співтовариством, організованим через Java Community Process, мова і основні реалізації цієї технології поширюються за ліцензією GPL.

Програми на Java транслюються в байт-код Java, який виконується віртуальною машиною Java (JVM) – програмою, яка обробляє байтовий код і передає інструкції обладнанню як інтерпретатор.

Перевагою подібного способу виконання програм є повна незалежність байт-коду від операційної системи і устаткування, що дозволяє виконувати Java-додатки на будь-якій платформі, для якої існує відповідна віртуальна машина. Іншою важливою особливістю технології Java є гнучка система безпеки, в рамках якої виконання програми повністю контролюється віртуальною машиною. Будь-які операції, які перевищують встановлені повноваження програми (наприклад, спроба несанкціонованого доступу до даних або з'єднання з іншим комп'ютером), викликають негайне переривання.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16





які дозволяють концентруватись на функціональних завданнях, в той час, як IntelliJ IDEA виконує рутинні робочі операції.

Продукт IntelliJ IDEA надає інтегрований інструмент для розробки графічного користувацького інтерфейсу. Серед інших можливостей, середовище добре інтегрується з багатьма популярними вільними інструментами розробки, такими як CVS, Subversion, Apache Ant, Maven та JUnit. У лютому 2007 року розробники IntelliJ анонсували ранню версію плагіну для підтримки програмування на мові Ruby.

Починаючи з версії 9.0, середовище доступне в двох редакціях: Community Edition та Ultimate Edition. Community Edition має повну вільну версію, доступну під ліцензією Apache 2.0, в ній реалізовано повну підтримку Java SE, Kotlin, Groovy, Scala, а також інтеграцію з найбільш популярними системами управління версіями. У редакціях Ultimate Edition, доступних під комерційною ліцензією, реалізована підтримка Java EE, UML-діаграм, підрахунок покриття коду, а також підтримка інших систем управління версіями, мовами та фреймворками.

IntelliJ IDEA 2020.1 підтримує нові можливості Java 14. IDE не тільки надає всі функції аналізу коду для записів (records), але і дозволяє швидко створювати нові записи, їх конструктори і компоненти, а також попереджає про помилки. Крім того, є підтримка зіставлення зі зразком для оператора instanceof, включаючи нову інспекцію і швидке виправлення, яке замінює надлишкові вирази instanceof + cast на лаконічні змінні-зразки.

Завантаження JDK через IDE. Більше не потрібно шукати необхідний JDK в інтернеті. При створенні проекту в IntelliJ IDEA 2020.1 можна скачати і встановити JDK прямо через IDE. При відкритті завантаженого проекту IDE перевірить, чи є на вашому комп'ютері відповідний JDK, і, якщо немає, запропонує його завантажити.

Нові інспекції для Java. З кожною новою версією IntelliJ IDEA розробники поширюють можливості статичного аналізу коду. Автори додали кілька нових інспекцій. Вони перевіряють коректність використання заголовних букв в

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

шаблонах SimpleDateFormat, аналізують випадки форматування дат, шукають надлишкові виклики методів, знаходять і пропонують видалити зайві екранують символи в строкових літералах.

У IntelliJ IDEA 2020.1 з'явився довгоочікуваний режим LightEdit. Тепер ви можете швидко вносити зміни в окремі файли в полегшеному редакторі, не створюючи й не завантажуючи проект. Відкривати файли в режимі LightEdit можна з командного рядка, з початкового екрана IDE або з файлового менеджера операційної системи.

Rename і Change Signature без модального діалогу. Щоб змінити сигнатуру методу або класу при розробці на Java або Kotlin, вам більше не доведеться мати справу з громіздким модальним діалогом. Просто змініть параметри класу або методу, а потім використовуйте новий значок на полі редактора або Alt + Enter, щоб переглянути і застосувати зміни до оголошення класу / методу і до всіх його використанням.

Розумна перевірка правопису. У IntelliJ IDEA 2020.1 вбудований плагін Grazie – універсальний інструмент для перевірки орфографії, граматики і стилю письма. Тепер IDE піклується не тільки про код. Плагін підтримує 16 природних мов і допоможе вам не допускати помилок в строкових літералах, коментарях, повідомленнях коммітів, Javadocs і т.д. За замовчуванням Grazie перевіряє тільки текстові файли, але ми рекомендуємо також включити перевірку Java-файлів: Settings / Preferences | Editor | Proofread | Grammar.

У отладчике JVM тепер підтримується аналіз потоків даних. Він пророкує, як буде виконуватися програма. Коли виконання програми переривається на точці зупинки, IDE робить аналіз потоків даних на основі поточного стану програми і перевіряє, що буде далі. Щоб включити аналіз, перейдіть в меню Settings / Preferences | Build, Execution, Deployment | Debugger | Data Views | Java.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

Профільювальник IntelliJ IDEA тепер виявляє рекурсивні виклики, коли один і той же метод викликається вище по стеку. IDE дозволяє прибрати такі виклики з подерева, щоб ви могли зосередитися на методах, які споживають більшу частину ресурсів, і на виклики, які вони виконують. Рекурсивні виклики відзначені спеціальним значком на вкладці Call Tree вікна Profiler.

### 2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи кібербезпеки для виявлення вразливостей web-застосунків.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

Кафедра КБПЗ – 2021 рік

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

## 3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

### 3.1 Опис функціонування системи

Число компаній, що застосовують веб-технології для підвищення продуктивності роботи й залучення нових клієнтів, росте з кожним роком. Безсумнівно, інтернет-сервіси несуть із собою безліч переваг, але є й зворотна сторона медалі – з ростом числа застосунків збільшується й кількість кіберзагроз. Так, компанія Symantec у своєму звіті Global Internet Security Threat Report (ISTR) указує, що кіберзлочинці при зломі веб-сайтів звичайно використовують уразливості веб-застосунків, що працюють на сервері, або експлуатують деякі уразливості операційної системи, на якій працюють ці застосунки. Наприклад, за допомогою атак типу XSS хакер може перенаправляти запити користувачів на зловливі веб-сторінки, а за допомогою SQL-ін'єкцій – витягати з баз даних сайту різну конфіденційну інформацію.

У відповідь на масові зломи систем безпеки був створений консорціум OWASP – Open Web Application Security Project, це відкритий проект забезпечення безпеки веб-застосунків. Однак і зловмисники, і фахівці в області кібербезпеки продовжують знаходити уразливості у веб-застосунках, які можуть привести до серйозних втрат з боку бізнесу. Основною причиною більшості проломів в Web-застосунках є написаний розроблювачами програмний код. Розроблювачі можуть допускати помилки при написанні коду або не усвідомлювати всю важливість використання приймань безпечного програмування – усе це приводить до появи уразливостей у застосунках.

Безсумнівно, захист веб-інфраструктури потрібна для будь-якої компанії. Однак з безлічі категорій захисних застосунків – Firewall, IPS/IDS, NGFW (Next Generation Firewall), WAF (Web-Application Firewall) тільки Web Application Firewall здатні забезпечити комплексний захист веб-застосунків від відомих і

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

невідомих загроз, а також забезпечити відповідність вимогам регуляторів, наприклад, стандартам PCI DSS. Не класичний Firewall, не IPS/IDS не зможуть забезпечити адекватного захисту веб-застосунків.

### **Еволюція брандмауерів**

По своїй суті файрволи (також застосовується термін брандмауер і міжмережевий екран) – це мережний фільтр між внутрішньою (корпоративною) і зовнішньою мережею (тобто, Інтернет-середовищем). Перші брандмауери були здатні лише блокувати підозрілі мережні пакети на мережному й каналному рівнях виходячи з IP -адреси джерела й призначення, прапора фрагментації й номера портів.

Більш просунуті системи – такі як IPS/IDS (Intrusion Prevention System/Intrusion Prevention System) – здатні аналізувати вміст мережних пакетів і порівнювати його із сигнатурами відомих атак. Крім того, ці системи виявляють і блокують відхилення в протоколах прикладного рівня.

Однак сьогодні понад 80% атак експлуатують уразливості застосунків, а не мережної архітектури. Тому вищеописана система захисту виявляється малоефективною проти сучасних кіберзагроз. До того ж, сьогодні існує величезна кількість веб-застосунків (кожне з яких потенційно може мати які-небудь уразливості), тобто, загальне число уразливостей набагато більше, ніж кількість сигнатур у базах сучасних Ips-систем. По оцінках експертів, саме проникнення через веб-застосунки останнім часом стають основним вектором атак на корпоративні мережі, причому традиційні системи безпеки, такі як файрвол і антивірусна система, ніяк не запобігали подібним атакам.

Для надійного захисту необхідний кардинально інший підхід: із глибоким аналізом змісту пакетів і гарним знанням структури веб-застосунків, включаючи Url-Параметри, cookie, форми введення даних і ін. Таким умовам задовольняє Web Application Firewall – брандмауер для застосунків, що здійснюють передачу даних через HTTP і HTTPS.

### **WAF на стражі інтернет-застосунків**

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

Варто акцентувати, що WAF – це вузькоспеціалізована система, яка аналізує тільки протоколи HTTP/HTTPS. З іншого боку, число способів обміну даними поверх протоколу HTTP настільки велике, що обробити їхній здатний тільки дійсно спеціалізований засіб. Крім того, багато файрволи для веб-застосунків підтримують роботу з SSL-трафіком. До речі, саме можливість перевірки зашифрованого трафіка є одним з найважливіших відмінностей WAF від звичайних міжмережєвих екранів і IPS, затверджують аналітики Gartner.

Для виявлення атак WAF застосовує як сигнатурний, так і поведінковий підходи. Другий метод також дуже важливий, оскільки для атак на веб-застосунки кіберзлочинці можуть застосовувати уразливості нульового дня (zero-day), що зводить до нуля ефективність сигнатурного аналізу. У той же час WAF здатний визначати модель нормального функціонування застосунки на основі аналізу мережного трафіка й системних журналів і її базі детектувати відхилення в поведінку програмної системи. Зокрема, WAF може виявляти атаки з використанням автоматичних засобів.

Класичний файрвол генерує величезні обсяги неправильних спрацьовувань на всілякі підозрілі події. Щоб детектувати рівень загрози, у цих повідомленнях треба розбиратися вручну. У той же час WAF здатний аналізувати тисячі подій і будувати ланцюжок розвитку атаки – від першого етапу до останнього.

### **Хто під прицілом?**

За даними щорічного дослідження корпоративних ризиків «Барометр ризиків Allianz 2021», яке було складено на основі опитування більш 820 ризик-менеджерів і страхових експертів з 44 країн, уперше в ТОП-3 корпоративних ризиків увійшли кіберзлочини. Вони ж вказуються як найбільш значний ризик для підприємств у довгостроковій перспективі наступних 10 років. Згідно із прогнозами дослідницької компанії Cybersecurity Ventures, до 2021 року збиток від кібершахрайства збільшиться вдвічі в порівнянні з 2020 роком і досягнеться \$6 трлн.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

Основною метою кіберзлочинців традиційно є різні фінансові організації, у першу чергу – банки, а також майданчика електронної комерції. При цьому, згідно даним звіту Trend Micro Incorporated за перше півріччя 2021 року, однією з найбільш значимих загроз у фінансовій галузі як і раніше залишаються банківські трояни. Украдена троянами інформація використовується правопорушниками для проведення шахрайських транзакцій або продається на підпільних сайтах. Більше того, у результаті подібних шкідливих дій фінансові організації змушено мати втрати на компенсацію збитків, які понесли їхні клієнти в результаті кібератак.

Дуже часто основною крапкою злому організації стає саме веб-застосунок. Загроза злому веб-застосунків залишається однією із самих серйозних проблем для веб-ресурсів будь-якої спрямованості. Відчути себе хакером сьогодні може навіть людина, слабо підготовлений технічно – методи злому й необхідні інструменти доступні у відкритому виді й легко можуть бути знайдені за допомогою звичайних розвідувачів.

### **Web Penetration Test**

Важливий організаційний момент при побудові системи захисту веб-застосунків – тест на проникнення, тобто, Penetration test. Саме він стане оптимальним способом перевірки захищеності інформаційної системи за допомогою імітації спрямованих атак. Penetration test дає можливість оцінити захищеність інформаційної системи від несанкціонованого впливу, використовуючи різні моделі вторгнень.

Тест на проникнення для веб-застосунків фокусується тільки на оцінці рівня захисту веб-застосунків. Процес складається з активного аналізу застосунків і пошуку в них уразливостей, технічних помилок або інших проблем. Інформація про всі слабкі місця відображається в підсумковому звіті.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

### 3.2 Розробка структурної схеми

На структурній схемі (рис. 3.1) представлено основні структурні особливості архітектури системи та відображено її головні структурні блоки:

- а) Інтерфейсний модуль;
- б) Модуль налаштування інтерфейсу та тестів;
- в) Модуль підключення та запуску тестових модулів;
- г) Модуль тестування вразливості операцій рівня адміністратора;
- д) Модуль тестування вразливості операцій рівня менеджменту;
- е) Модуль тестування вразливості операцій рівня користувача;
- ж) Модуль тестування безпеки фінансових операцій.

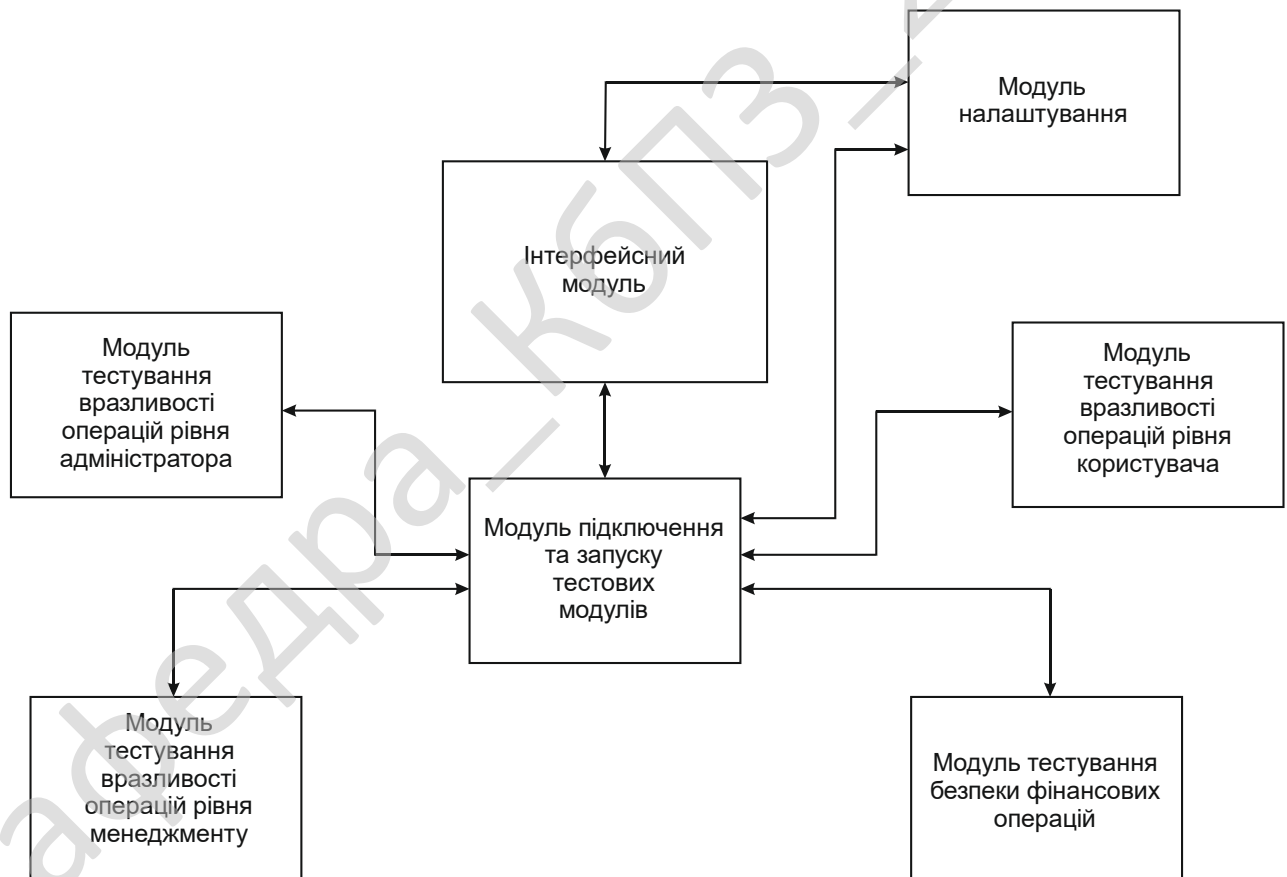


Рисунок 3.1 - Структурна схема роботи системи

### 3.3 Розробка функціональної схеми

Функціональну схему розробленого програмного забезпечення та системних складових відображено на рис 3.2.

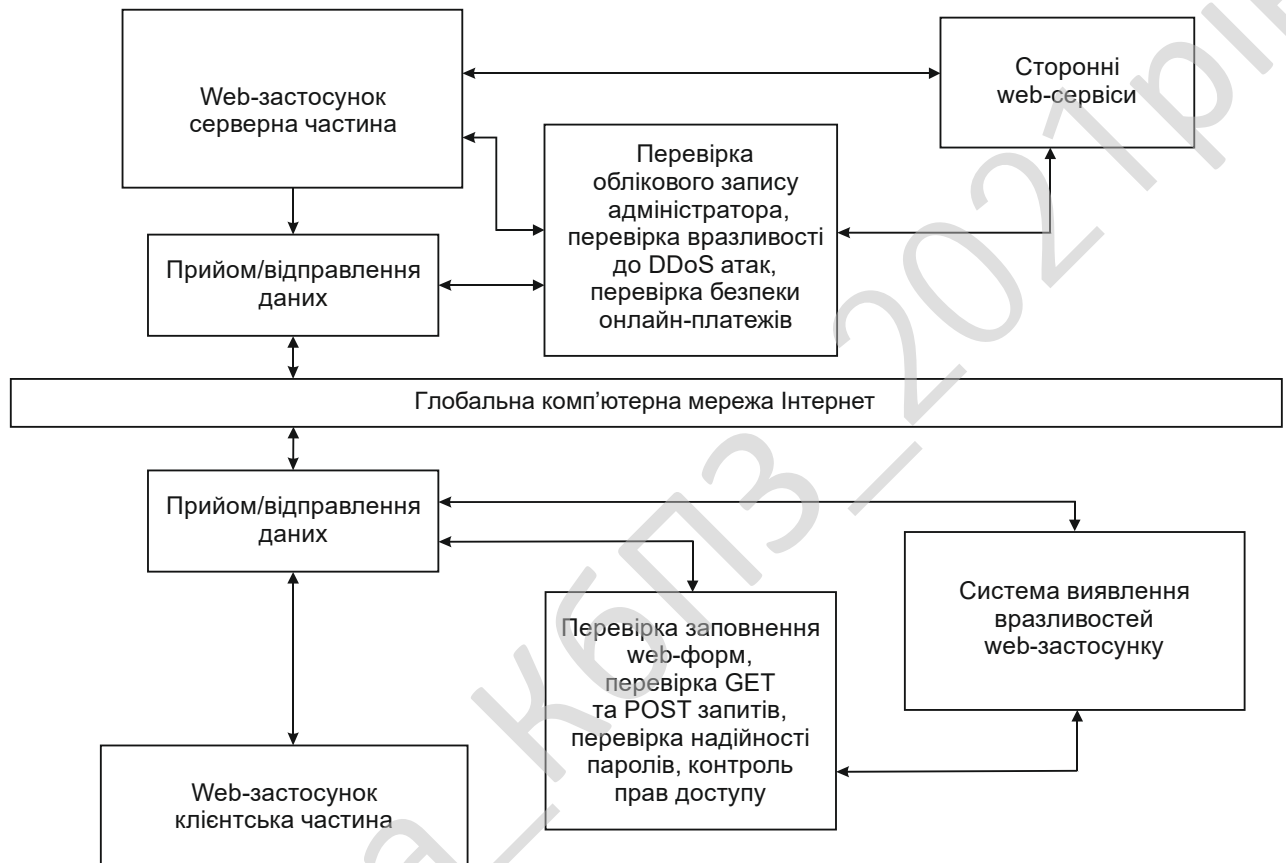


Рисунок 3.2 – Функціональна схема системи

Після запуску програми виконується перевірка та запуск наявних тестів. Інформація про виявленні вразливості зберігається у вигляді текстових лог-файлів.

### 3.4 Розробка діаграми процесів

Чітка синхронізація та ієрархія процесів є важливою складовою при розробці програмного забезпечення, тому до початку розробки необхідно розробити діаграму взаємодії процесів (рис. 3.3).

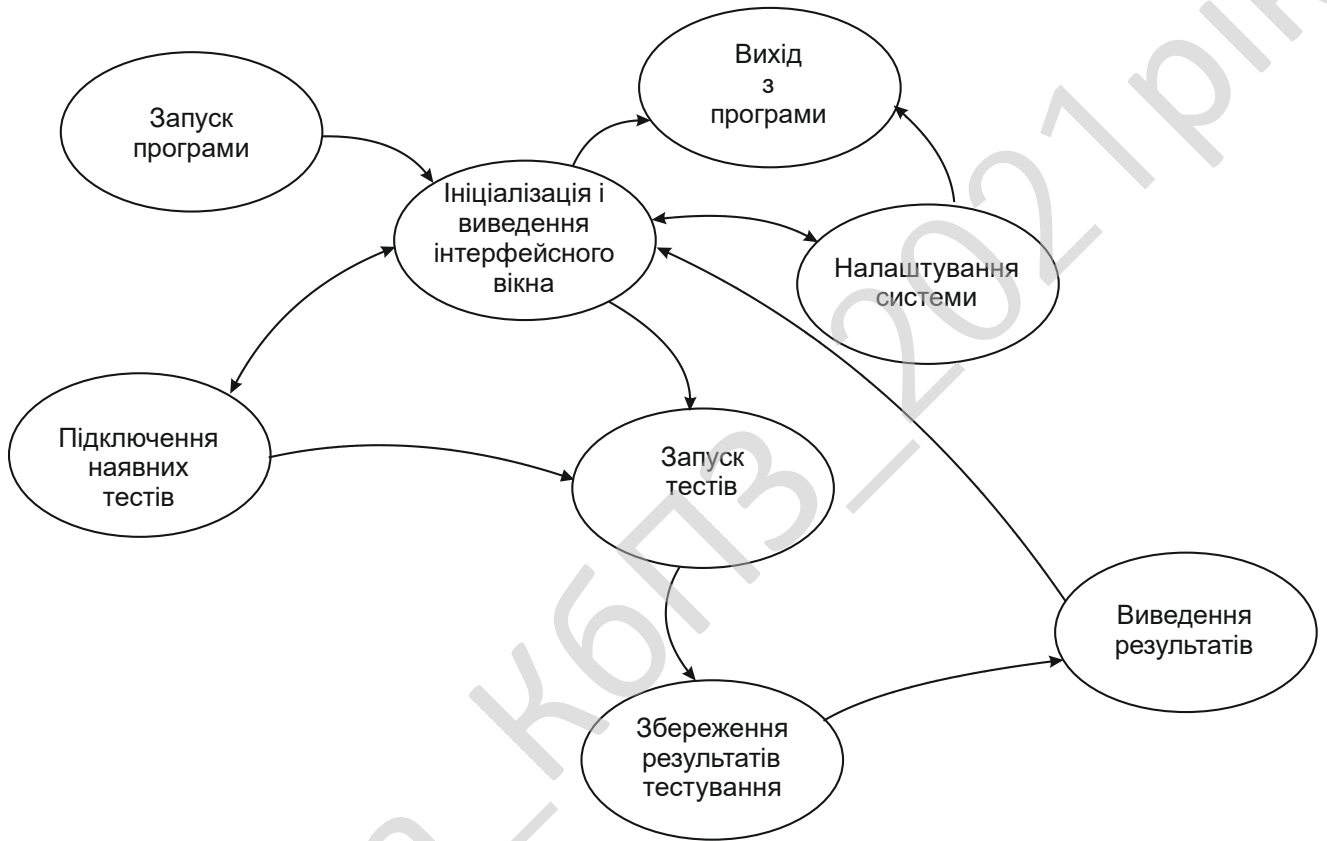


Рисунок 3.3 – Діаграма взаємодії процесів

Після етапу аналізу та підготовки структури і функціональних вимог до системи починаємо розробку алгоритмів системи та безпосередньо до розробки програмного коду.

## 4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ ТА ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ І ПРОГРАМНИХ РІШЕНЬ

### 4.1 Розробка блок-схем та опис алгоритмів функціонування системи

На рисунку 4.1 відображено загальний алгоритм програми. Після підключення всіх наявних тестів виконується їх поетапний запуск. Після виконання кожного тесту інформація про виконання зберігається в лог-файл. У випадку, якщо тест не пройдено (через тайм-аут, відсутність зв'язку з сервером, некоректні вхідні дані) тест переривається і в лог-файл зберігається інформація про помилку.

На блок-схемах алгоритмів підпрограм (рис. 4.2 та 4.3) відображено процес виконання перевірки стійкості нового паролю та системи автентифікації та збереження сесії користувача.

Відповідно до вимог безпеки новий пароль перевіряється на кількість символів (не менше 9), наявність великих символів, цифр та спец-символів, така висока складність паролю обумовлена наявністю в web-застосунку виконанням операцій онлайн-оплат.

Перевірка системи автентифікації та збереження сесії користувача необхідна для виключення ситуацій, коли користувач не виконавши операцію виходу з облікового запису може залишити свій обліковий запис вразливим.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

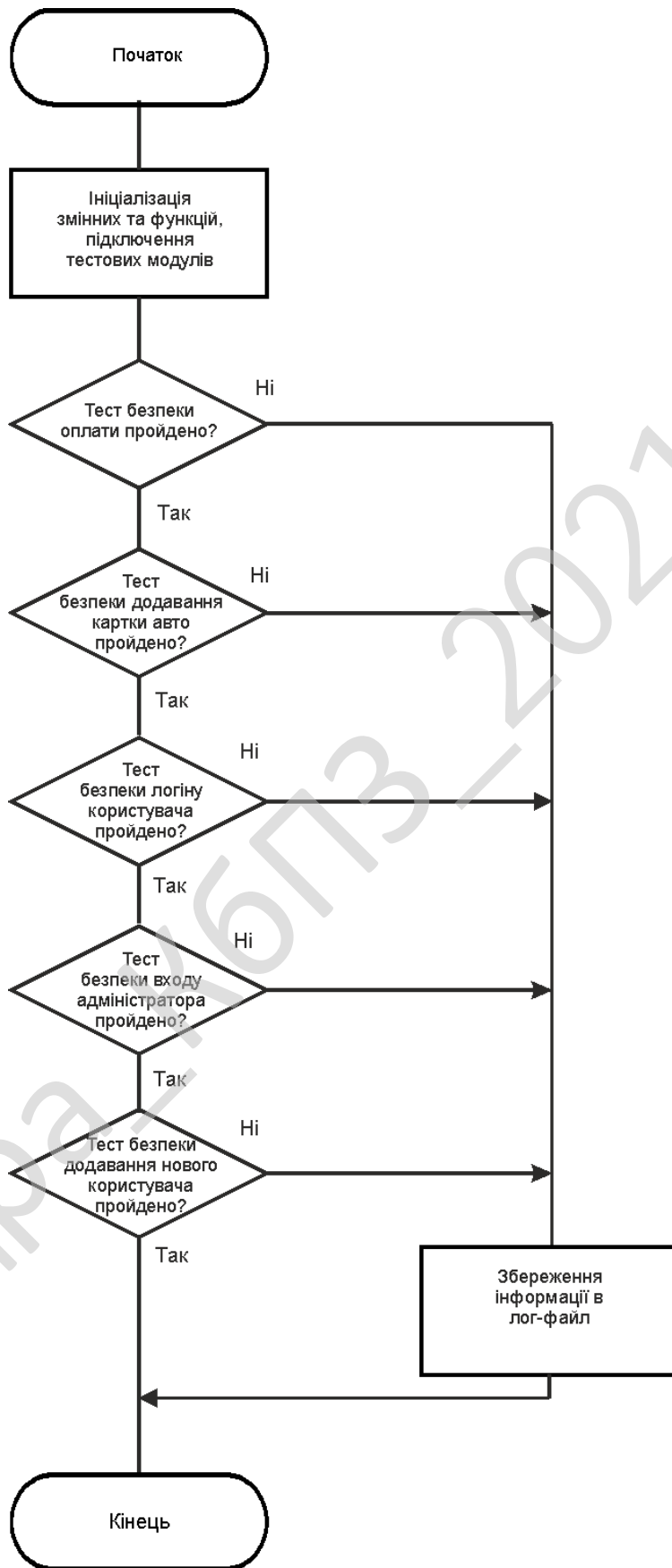


Рисунок 4.1 – Блок-схема алгоритму роботи додатку

Вим.	Арк.	№ докум.	Підпис	Дата

ВКРМ-123.21.0022.00.00.ПЗ

Арк.

31

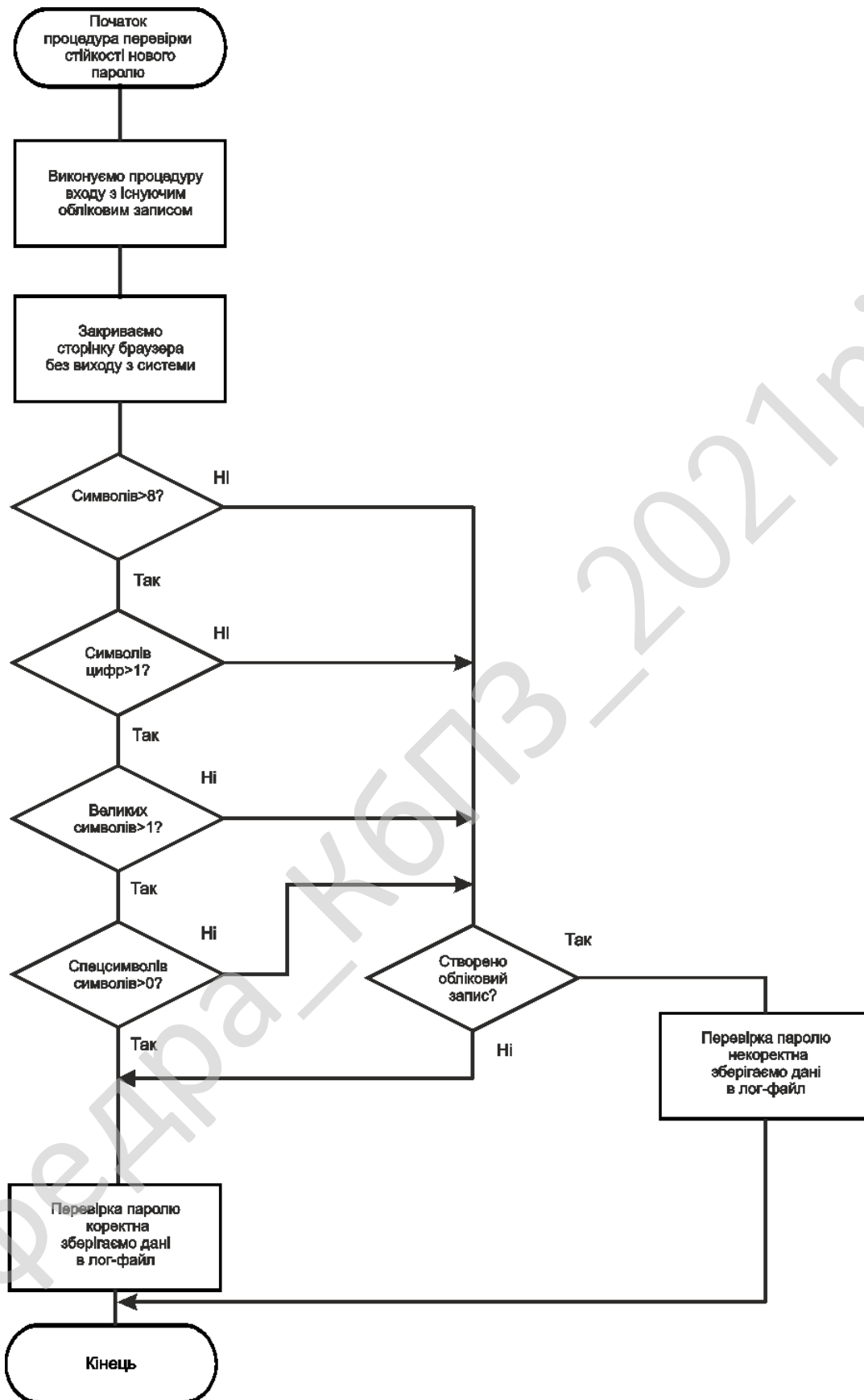


Рисунок 4.2 – Блок-схема алгоритму роботи процедури перевірки стійкості нового паролю

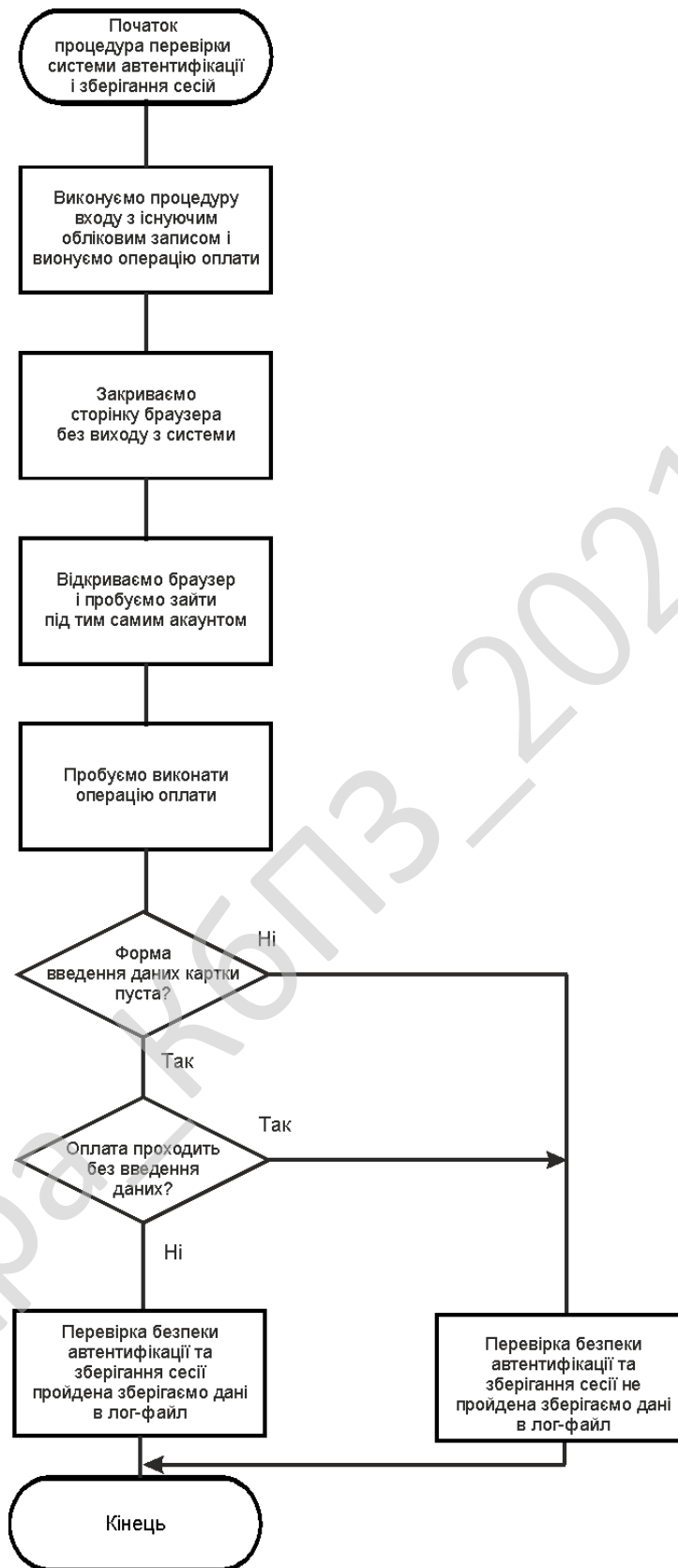


Рисунок 4.3 – Блок-схема алгоритму роботи процедура перевірки системи автентифікації і зберігання сесій.

Програмне забезпечення проекту реалізовано, як набір окремих підключаємих тестових модулів (процедур).

Головний модуль призначений для загального налаштування системи та підключення і запуску тестових модулів, керування роботою модулів і компонентів системи, а також обробки помилок, якщо вони з'являться в ході роботи. У випадку виникнення помилки (наприклад відсутній вказаний тестовий модуль або відсутній зв'язок з сервером) виводиться діагностична інформація, якщо помилка критична, то додаток аварійно завершується.

В якості прикладу використання системи було тестове web-застосування сайту оренди автомобілів <https://beta.t.tcicarrentalengines.us>. Для тестування операцій оплати використано сервіс онлайн-оплат Stripe.

Отже спочатку відбувається підключення і запуск тестових модулів:

```
import org.junit.platform.runner.JUnitPlatform;
import org.junit.platform.suite.api.SelectClasses;
import org.junit.runner.RunWith;
```

```
@RunWith(JUnitPlatform.class)
@SelectClasses({
    TestSuiteCreateRent.class,
    TestSuiteCreateUser.class,
    TestSuiteCreateCar.class
})
public class RunAllTestSuits {}
```

Перевірка коректності заповнення форми оплати за послугу оренди автомобіля:

```
RentFormTestData rentTestData = new RentFormTestData(
    carEntity.carType + "", "1",
    new DateWorker().getTodayPlusDates(1) + "",
    new DateWorker().getTodayPlusDates(2) + "",
    "test name", "test@4fnerore.com", "4395494554", "test addr"
);

LandingPage.RentForm form = landingPage.getRentForm();

form.getErrorMessage().shouldBe(not(visible));

form.getCarType().select(rentTestData.carType);
form.getGuestsCount().select(rentTestData.guests);
form.getFrom().select(rentTestData.From);
form.getTo().select(rentTestData.To);

form.getPlanInfo().getSedanPlan().check("Insurance included", price);
form.getPlanInfo().getNext()
    .check(price)
    .click();

form.getUserInfo().fill(
```

										Арк.
										34
Вим.	Арк.	№ докум.	Підпис	Дата	ВКРМ-123.21.0022.00.00.ПЗ					

```

        rentTestData.username, rentTestData.email,
        rentTestData.phone, rentTestData.address
    );
    form.getBookResult()
        .check("Insurance included", price, 1, "Taxes are included")
        .getBookNow().click();

```

## Перевірка операції бронювання та внесення передплати

```

PaymentPage paymentPage = new PaymentPage();
paymentPage.getReservationDetails().check(
    rentTestData.username, rentTestData.phone, rentTestData.address,
    rentTestData.From, rentTestData.To, rentTestData.guests
);
paymentPage.getPaymentDetails()
    .check(price, true)
    .getPurchase().click();

StripePaymentPopup popup = StripePaymentPopup.render()
    .fill(rentTestData.email, "4242 4242 4242 4242", "022022", "222")
    .check(
        carEntity.name + "", rentTestData.email + "",
        "4242 4242 4242 4242", "02 / 2022", "222"
    );
popup.getPayWithCard().click();

SuccessPaymentPage successPaymentPage = SuccessPaymentPage.render()
    .check(
        rentTestData.username + "",
        rentTestData.From + "",
        rentTestData.To + "",
        rentTestData.carType.replace(" (Diesel)", "") + "",
        "Insurance included",
        rentTestData.guests + "",
        price + "",
        rentTestData.email + ""
    );

```

## Перевірка операції додавання в систему нового користувача:

```

@DisplayName("TEST CASE: Create new client (STRIPE, Without Source)")
@Order(2)
@Test
public void createNewClientStripeWithoutSource() {
    AdminLoginPage.open()
        .login("default-admin@mail.com", "default");

    new AdminSidebar().getAddNewClient().click();
    AddNewClientPopup popup = AddNewClientPopup.render();
    popup.fill(
        "Devbug2" + code,
        "test2_" + code.replace(" ", ""),
        email + "",
        "95495423949",
        "test google 2 (en)" + code,
        "test google 2 (jp)" + code,
        "Without source",
        popup.fillStripeData("10"),
        true
    );

```

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

```

).getAddNewClient().click();

SingUpNewClientEmail mail = new SingUpNewClientEmail(Objects
    .requireNonNull(MailController.waitNewMail(120)));
mail.check(email, "test2_" + code.replace(" ", ""));

CompleteSingUpPage completeSingUpPage =
CompleteSingUpPage.open(mail.getInviteLink());
String password = System.getenv("TcicarrentalRentTestPass");
completeSingUpPage
    .fill(password + "_Te$1", password + "_Te$1")
    .getSetPassword().click();
new CarsPage();
new PageHeader().getLogout().click();
}

```

### Процедура перевірки операції додавання в систему картки нового автомобіля:

```

System.getenv("TCIApiKeyPass").getConfirm().click();
Selenide.sleep(3000);

String code = " #" + String.format("%06d", new Random().nextInt(999999));

CarNewPage carNewPage = new CarNewPage();
CarNewPage.RegisterCar registerCar = carNewPage.getRegisterCar();
registerCar.fill(
    "Parallel Palms Car (en) [auto]" + code,
    "Parallel Palms Car (en) [auto]" + code,
    "P48909",
    new File("src/main/resources/" + "images/logos/car_logo.useg"),
    new File("src/main/resources/" +
"images/exterior/exterior_car.usg"),
    "0100973", "Ford",
    "4-chōme-6 Ford (en)",
    "4-chōme-6 Ford (en)",
    "5495959444",
    "Privacy policy (en)", "Privacy policy (en)"
);
registerCar.getSaveAndAddCancellationPolicy().click();

CarNewPage.CancellationSettings cancellationSettings =
carNewPage.getCancellationSettings();
cancellationSettings.fillEarlyCancellationFeeSettings(
    "3", "5", "10"
);
cancellationSettings.getSaveAndAddCancellationPolicy().click();

CarNewPage.AddCar addCar = carNewPage.getAddCar();
addCar.fill(
    "Double (en)" + code,
    "Double (en)" + code,
    false,
    "Double",
    "4",
    "2",
    true,
    "17074713",
    false,
    null,

```

					<b>БКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

```

        new ArrayList<>(Arrays.asList("TV", "Comb")), // these amenities
will be exclude
        "Car description (en)",
        "Car description (en)",
        new File("src/main/resources/" + "images/vip_car/vip_car_1.usg"),
        new File("src/main/resources/" + "images/vip_car/vip_car_2.usg")
    );
    addCar.getComplete().click();

```

**Процедура перевірки вразливості облікового запису адміністратора  
реалізована наступним чином:**

```

public class AdminLoginPage {
    private LoginForm loginForm;

    public AdminLoginPage() {
        render();
    }

    @Step("open [Admin Login page]")
    public static AdminLoginPage open() {
        return Selenide.open(AdminLoginPage.url(), AdminLoginPage.class);
    }

    public static String url() {
        return Domen.ADMIN.getUrl().concat("login");
    }

    public void render() {
        if (getStorage().getItem("locale") == null) {
            getStorage().setItem("locale", "en");
            Selenide.refresh();
        }
        this.loginForm = new LoginForm();
        loginForm.render();
    }

    public LoginForm getLoginForm() {
        return loginForm;
    }

    public void login(String email, String password) {
        getLoginForm().getEmail().setValue(email);
        getLoginForm().getPassword().setValue(password);
        getLoginForm().getSubmit().click();
        new ClientsPage();
    }
}

```

**Перевірка операцій оплати оренди авто:**

```

payment).legalInformationJpData.representative,
        ((StripeData) payment).legalInformationJpData.address,
        ((StripeData) payment).legalInformationJpData.sellingPrice,
        ((StripeData)
payment).legalInformationJpData.additionalSurcharges,

```

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

```

        ((StripeData)
payment).legalInformationJpData.paymentPeriodAndMethod,
        ((StripeData) payment).legalInformationJpData.deliveryTime,
        ((StripeData) payment).legalInformationJpData.returns,
        ((StripeData) payment).legalInformationJpData.cancellation,
        ((StripeData) payment).legalInformationJpData.phoneNumber,
        ((StripeData) payment).legalInformationJpData.email
    );
    if (((StripeData) payment).legalInformationEnData != null) {
        getNext().click();
        renderLegalInformationEn();
        getLegalInformationEn().fill(
            ((StripeData) payment).legalInformationEnData.seller,
            ((StripeData)
payment).legalInformationEnData.representative,
            ((StripeData) payment).legalInformationEnData.address,
            ((StripeData)
payment).legalInformationEnData.sellingPrice,
            ((StripeData)
payment).legalInformationEnData.additionalSurcharges,
            ((StripeData)
payment).legalInformationEnData.paymentPeriodAndMethod,
            ((StripeData)
payment).legalInformationEnData.deliveryTime,
            ((StripeData) payment).legalInformationEnData.returns,
            ((StripeData)
payment).legalInformationEnData.cancellation
        );
        getSendInvitationEmail().setFlag(isSendInvitationEmail);
    }
}

```

#### 4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Blowfish, який є симетричним алгоритмом шифрування, тобто таким, у якому ключ шифрування дорівнює ключу дешифрування. Він є мережею Фейштеля, у якій кількість ітерацій дорівнює 16. Довжина блоку дорівнює 64 бітам, ключ може мати будь-яку довжину в межах 448 біт. Хоча перед початком будь-якого шифрування виконується складна фаза ініціалізації, саме шифрування даних виконується досить швидко.

Алгоритм призначений в основному для додатків, у яких ключ міняється нечасто, до того ж існує фаза початкового рукостискання, під час якої відбувається автентифікація сторін і узгодження загальних параметрів і секретів. При реалізації на 32-бітних мікропроцесорах з більшим кешем даних Blowfish значно швидше DES.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

Алгоритм складається із двох частин: розширення ключа й шифрування даних. Розширення ключа перетворює ключ довжиною, принаймні, 448 біт у кілька масивів підключів загальною довжиною 4168 байт.

В основі алгоритму лежить мережа Фейштеля з 16 ітераціями. Кожна ітерація складається з перестановки, що залежить від ключа, і підстановки, що залежить від ключа й даних. Операціями є XOR і додавання 32-бітних слів.

Blowfish використовує велику кількість підключів. Ці ключі повинні бути обчислені заздалегідь, до початку будь-якого шифрування або дешифрування даних. Елементи алгоритму:

1.  $P$  – масив, що складається з вісімнадцяти 32-бітних підключів:

$P_1, P_2, \dots, P_{18}$ .

2. Чотири 32-бітних  $S$ -boxes с 256 входами кожний. Перший індекс означає номер  $S$ -box, другий індекс – номер входу.

$S_{1,0}, S_{1,1}, \dots S_{1,255};$

$S_{2,0}, S_{2,1}, \dots S_{2,255};$

$S_{3,0}, S_{3,1}, \dots S_{3,255};$

$S_{4,0}, S_{4,1}, \dots S_{4,255};$

### Шифрування

Входом є 64-бітний елемент даних  $X$ , що ділиться на дві 32 -бітні половини,  $X_l$  і  $X_r$ .

$$X_l = X_l \text{ XOR } P_i$$

$$X_r = F(X_l) \text{ XOR } X_r$$

Swap  $X_l$  and  $X_r$

### Функція $F$

Розділити  $X_l$  на чотири 8-бітних елементи  $A, B, C, D$ .

$$F(X_l) = ((S_{1,A} + S_{2,B} \text{ mod } 2^{32}) \text{ XOR } S_{3,C}) + S_{4,D} \text{ mod } 2^{32}$$

Дешифрування відрізняється від шифрування тим, що  $P_i$  використовуються у зворотному порядку.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

### Генерація підключів

Підключи обчислюються з використанням самого алгоритму Blowfish.

1. Ініціалізувати перший  $P$ -масив і чотири  $S$ -boxes фіксовані рядки.

2. Виконати операцію XOR  $P_1$  з першими 32 бітами ключа, операцію XOR  $P_2$  із другими 32 бітами ключа й т.д. Повторювати цикл доти, поки весь  $P$ -масив не буде побітово складний з усіма бітами ключа. Для коротких ключів виконується конкатенація ключа із самим собою.

3. Зашифрувати нульовий рядок алгоритмом Blowfish, використовуючи підключи, описані в пунктах (1) і (2).

4. Замінити  $P_1$  і  $P_2$  виходом, отриманим на кроці (3).

5. Зашифрувати вихід кроку (3), використовуючи алгоритм Blowfish з модифікованими підключами.

6. Замінити  $P_3$  і  $P_4$  виходом, отриманим на кроці (5).

7. Продовжити процес, замінюючи всі елементи  $P$ -масиву, а потім всі чотири  $S$ -boxes, виходами відповідним чином модифікованого алгоритму Blowfish.

Для створення всіх підключів потрібна 521 ітерація.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ док.ум.	Підпис	Дата		40



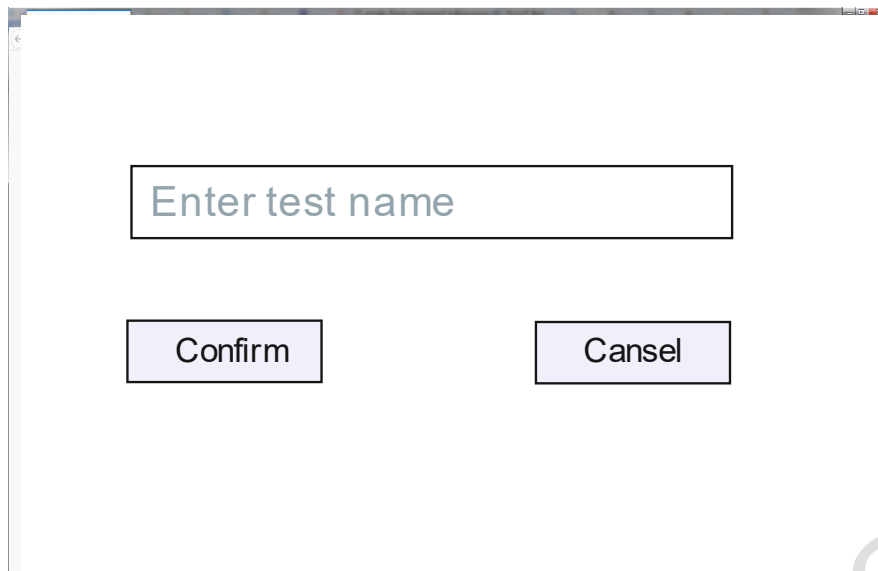


Рисунок 5.2 – Інтерфейс додавання нового тесту для виявлення вразливостей web-застосунків в систему

Інформація про результати виконання тестів додається в текстовий файл testresult.log в кореневому каталозі проекту, якщо файлу не знайдено, то створюється новий.

## 6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи кібербезпеки для виявлення вразливостей web-застосунків.

*Метою розробки є дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.*

*Об'єктом дослідження є процес кібербезпеки для виявлення вразливостей web-застосунків.*

*Предметом дослідження є методи кібербезпеки для виявлення вразливостей web-застосунків.*

*Методи дослідження базуються на методах теорії захисту інформації, методах математичної статистики, методах розробки програмного забезпечення.*

**Наукова новизна отриманих результатів.** У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод кібербезпеки для виявлення вразливостей web-застосунків.

– Розроблено вітчизняний продукт кібербезпеки для виявлення вразливостей web-застосунків, який має більш широкі можливості, на відміну від існуючих аналогів.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

## 7 ДАНІ ПРО ЕКОНОМІЧНУ ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ

### 7.1 Техніко-економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Після ознайомлення з підприємством та засобами розробки програмної продукції був розроблений план розробки програми. Був підрахований необхідний час для розробки та впровадження програми. Цей час склав 60 днів (три місяці). В магістерській роботі було проведене дослідження та виконана програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

Розроблене програмне забезпечення має достатню надійність і задовольняє усім поставленим умовам, а саме:

- а) невеликий розмір;
- б) невеликі системні потреби;
- в) незалежність від встановлених на комп'ютері баз даних;
- г) зручність у користуванні та надійність.

Таблиця 7.1 – Початкові дані

Показники	Позначення	Характеристика або величина
1	2	3
1. Кількість розроблених програм період, шт.	N	1
2. Кількість екземплярів програм, шт.	№	22 (2 ост. цифри № зал)
3. Запланований термін розробки, днів	Frq	60 (3 місяці)
4. Група задачі підсистеми управління (1-6)	–	1
5. Ступінь новизни задачі (А, Б, В, Г)	–	Б
6. Складність алгоритму (1, 2, 3)	–	2

Продовження таблиці 7.1

1	2	3
7. Кількість макетів вхідної інформації	–	3
8. Кількість форм вихідної інформації.	–	4
9. Мова програмування (1-6)	–	3
10. Попередній досвід (1-6)	–	3
11. Гнучкість проекту ПП (1-6)	–	3
12. Детальність проекту ПП (1-6)	–	2
13. Рівень спрацьованості колективу (1-6)	–	2
14. Ступінь вимірності процесів (1-6)	–	3
15. Необхідна надійність програмного забезпечення (1-6)	–	2
16. Розмір бази даних (порівняно з розміром програми) (1-6)	–	2
17. Складність кінцевого програмного продукту (1-6)	–	2
18. Необхідний рівень забезпечення повторного використання (1-6)	–	2
19. Документованість відповідно до планованого життєвого циклу (1-6)	–	2
20. Вимоги до швидкодії ПП (1-6)	–	2
21. Обмеження на розміри основного сховища даних (1-6)	–	2
22. Різноманітність використовуваних обчислювальних платформ (1-6)	–	2
23. Професійний рівень аналітиків (1-6)	–	2
24. Професійний рівень програмістів (1-6)	–	2
25. Постійність складу команди розробників (1-6)	–	2
26. Досвід розробки додатків (1-6)	–	2
27. Досвід роботи з обчислювальною платформою (1-6)	–	2

Вим.	Арк.	№ докум.	Підпис	Дата

ВКРМ-123.21.0022.00.00.ПЗ

Арк.

45

Продовження таблиці 7.1

1	2	3
28. Досвід роботи з мовою і інструментами середовища розробки (1-6)	–	2
29. Досвід роботи з програмними інструментами розробки (1-6)	–	3
30. Розробка ПЗ для декількох серверів одночасно (1-6)	–	2
31. Вимоги до дотримання встановленого графіка робіт (1-6)	–	2
32. Вартість ПЗ у розробника (НМА), грн.	–	22000
33. Норматив додаткової зарплати, % :	Нд	10
34. Норматив відрахувань у соціальні фонди, %	Нс	22
35. Норматив загальногосподарських витрат, %	Нг	15
36. Норматив витрат на освоєння нових мов програмування, %	Нп	15
37. Рівень рентабельності програмної продукції, %	Ре	50
38. Ставка податку на додану вартість, %	Ндв	20

## 7.2 Розрахунок трудомісткості розробки програмної продукції

Значення трудомісткості розробки програмного забезпечення для стадій ТЗ, ЕК, ТП та ВП визначаємо по типовим нормам часу приведеним в додатках МВ. Стадія РП є найбільш тривалою і трудомісткою, що робить значний вплив на інші стадії проекту.

Визначимо трудомісткість розробки ПЗ для стадії РП.

Обчислюємо номінальні трудовитрати, люд-міс.:

$$T_{ном} = A \text{ Size}^B, \quad (7.1)$$

де:  $A$  – коефіцієнт Боема,  $A = 2,45$ ;

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		46

Size – загальний об'єм відлагодженого програмного коду, тис. рядків;

$B$  – показник ступеня, що визначається співвідношенням:

$$B = 1,01 + 0,001 \sum W_i, \quad (7.2)$$

де:  $W_i$  – сумарне значення п'яти показників (МВ, додаток 2), що відображають особливості розробки проекту програмного продукту (ПП) і колективу розробників.

$$B = 1,01 + 0,001(2,43 + 3,64 + 3,38 + 3,95 + 2,73) = 1,027.$$

$$T_{ном} = 2,45 \cdot 2,7^{1,026} = 6,78 \text{ люд-міс.}$$

Визначаємо уточнені (з урахуванням приведених в МВ додатку 3 сімнадцяти додаткових коефіцієнтів) трудовитрати, люд-міс.:

$$T_{уточн} = T_{ном} \prod V_j, \quad (7.3)$$

де:  $\prod V_j$  – добуток сімнадцяти додаткових коефіцієнтів, приведених в МВ додатку 3.

$$T_{уточн} = 6,78 \cdot (0,88 \cdot 0,93 \cdot 0,88 \cdot 0,91 \cdot 0,95 \cdot 1 \cdot 1 \cdot 0,87 \cdot 1,22 \cdot 1,16 \cdot 1,1 \cdot 1,1 \cdot 1,12 \cdot 1,1 \cdot 1,1 \cdot 1,1) = 9,37 \text{ люд-міс.}$$

Ці коефіцієнти дозволяють диференційовано оцінювати результати роботи програмістів, беручи до уваги швидкодію програми, використання різноманітних обчислювальних платформ і інструментів розробки, взаємодію декількох серверів, вимоги до об'ємів баз даних і ін.

Визначаємо підсумкові трудовитрати по стадії робочий проект, люд-дні:

$$T_{РП} = 0,3 C T_{уточн}^{0,33+0,2(B-1,01)} S, \quad (7.4)$$

де:  $C$  – визначений емпірично коефіцієнт, запропонований авторами методики, (МВ, додаток 4);

$S$  – коефіцієнт стиснення (або подовження) графіка робіт %, що дозволяє коректувати терміни розробки ПЗ згідно встановленим вимогам. Вибираємо в межах (25...350)%.

$$T_{РП} = 0,3 \cdot 2,85 \cdot 9,37^{0,33+0,2(1,026-1,01)} \cdot 80 = 168 \text{ люд/день.}$$

Для зручності визначення загальної трудомісткості на розробку програмного забезпечення результати розрахунків по стадіям зводимо до таблиці 7.2.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

Таблиця 7.2 – Визначення трудомісткості розробки програмного забезпечення

Стадії розробки	Трудомісткість за типовими нормами та розрахунками	
	Величина, люд/дні	Підстава
Технічне завдання	9	Д5
Ескізний проект	10	Д6
Технічний проект	9	Д7
Робочий проект	168	Ф 7.1-7.4
Впровадження	13	Д13
Всього	209	–

### 7.3 Визначення чисельності виконавців і планового фонду зарплати

Чисельність ставок інженерів-програмістів для розробки програмного забезпечення визначається за формулою:

$$Ч = \frac{T_{nz} N}{F_{pq} - H_{ev}}, \quad (7.5)$$

де:  $F_{pq}$  – плановий фонд робочого часу одного спеціаліста, днів;

$T_{nz}$  – трудомісткість розробки програмного забезпечення люд-дні.

$$Ч = \frac{209 \cdot 1}{60 - 5} = 3,8 \text{ ставки.}$$

Чисельність інженерів-електронщиків для проведення технічного обслуговування та ремонту комп'ютерних мереж визначається в залежності від наявності технічних засобів і норм витрат часу на виконання профілактичних робіт на протязі року.

Визначаємо затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за період розробки. Результати розрахунку зводимо до таблиці 7.3.

Таблиця 7.3 – Затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за розрахунковий період

Найменування обладнання	Профілактичне обслуговування			
	Кількість хв. на один. обл.	Кількість обладнання	Затрати часу в хв.	Затрати часу в год.
Системний блок ПК	385	12	4620	77
Монітор	160	12	1920	32
Клавіатура	140	12	1680	28
Маніпулятор «мишка»	30	12	360	6
Принтер матричний	185	1	185	3
Принтер лазерний	355	2	710	12
Принтер струминний	300	1	300	5
Сканер	155	2	310	5
Концентратор-маршрутизатор	155	2	310	5
Кабельні господарства ЛОМ на 1 м. п.	2,5	100	250	4
Кабельне господарство електромережі	48	50	2400	40
Копіювальний апарат	285	2	570	10
Усього за рік:			3 <sub>ч</sub>	227

Час на профілактику обладнання в загальному балансі робочого часу інженерів-електронщиків не повинен складати більше 10%.

Виходячи з цього фонд робочого часу інженерів-електронщиків складає:

$$\Phi_{ор}^c = \frac{3_{ч} \cdot n_{міс}}{1,2}, \quad (7.6)$$

$$\Phi_{ор}^c = \frac{227 \cdot 3}{1,2} = 567,5 \text{ год.}$$

Визначаємо необхідну кількість ставок штатного персоналу сектора ТО:

$$Ч_{ел} = \frac{\Phi_{др}^c}{F_{др} \cdot T_{зм}}, \quad (7.7)$$

$$Ч_{ел} = 567,5 / (60 \cdot 8) = 1,2 \text{ ставки.}$$

Для забезпечення нормального технічного обслуговування засобів ТО та мереж, необхідно прийняти найбільше ціле значення розрахункової чисельності інженерів-електронщиків.

Чисельність інженерів-системотехніків, адміністраторів мережі, дизайнерів WEB вузлів, системних програмістів (аналітиків), бухгалтерів-економістів визначається за потребою в залежності від функціональних обов'язків. Після визначення чисельності персоналу складається штатний розклад.

Таблиця 7.4 – Розрахунок чисельності штатного персоналу сектору системного та адміністративного обслуговування засобів ОТ та комп'ютерних мереж

Посада	Вид роботи	Час	К-ть штатних одиниць
Адміністратор загальної мережі, аналітик	Адміністрування локальної мережі, поштового та серверу DNS (OC FreeBSD), маршрутизатора Cisco, доменного контролеру Windows Server, серверу доступу ADSL (OC Linux), налаштування ADSL, VPN PPPoE, Frame Relay, Wi-Fi	2	0,5
	Налаштування і конфігурування базової станції безпроводного зв'язку (CMTS)	0,5	
	Розробка та впровадження проектів з організації зв'язку між віддаленими об'єктами, ЛОМ	0,5	
	Забезпечення цілодобової роботи зв'язку клієнтів до мережі Інтернет	1	
Всього		4	

Продовження таблиці 7.4

Посада	Вид роботи	Час	К-ть штатних одиниць
Продакт-менеджер	Презентації нової продукції, пошук каналів збуту	1	0,25
	Підтримка постійних клієнтів	0,5	
	Оформлення договорів, ведення тендерів	0,25	
	Контроль взаєморозрахунків з постачальниками	0,25	
Всього		2	
Дизайнер WEB	Розробка концепції оформлення та інтерфейсу сайту, оптимізація дизайну існуючих, проектує їх структуру та навігацію	1	0,25
	Створення графічних і стилістичних елементів сайту	0,5	
	Оформлення банерів і промо-сторінок	0,25	
	Розміщення графіки і контенту на Інтернет сторінках	0,25	
Всього		2	
Інженер верстальник	Розробка та верстка макетів рекламної продукції та технічної документації	1	0,25
	Верстка друкованих видань	0,5	
	Додрукова підготовка макетів	0,25	
	Розміщення графіки і контенту на Інтернет сторінках	0,25	
Всього		2	

Вим.	Арк.	№ докум.	Підпис	Дата
------	------	----------	--------	------

ВКРМ-123.21.0022.00.00.ПЗ

Арк.

51

Складемо штатний розклад виконавців.

Таблиця 7.5 – Штатний розклад виконавців

Посада	Кількість ставок	Середньомісячний оклад, грн.	Всього за період розробки, грн.
Керівник (ІТ-менеджер)	1	13000	39000
Продакт-менеджер	0,25	12000	9000
Інженер-програміст	3,8	12000	136800
Інженер-електронщик	1,2	11500	41400
Інженер-системотехнік	0,25	11500	8625
Адміністратор мережі	0,5	11500	17250
Системний програміст	0,25	11500	8625
Дизайнер WEB	0,25	12000	9000
Інженер-верстальник	0,25	11700	8775
Бухгалтер-економіст	0,5	12500	18750
Всього за період розробки	$R_{cn} = 8,25$	-	$\Phi_{роб} = 297225$

Розрахуємо середньоденну зарплату одного виконавця:

$$z_{cd} = \frac{\Phi_{роб}}{R_{cn} F_{pq}}, \quad (7.8)$$

де:  $\Phi_{роб}$  – загальна сума зарплати за плановий період, грн.

$$z_{cd} = \frac{297225}{8,25 \cdot 60} = 600 \text{ грн.}$$

#### 7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника

Балансова вартість будівель визначається з урахуванням кількості робочих місць виконавців, питомої площі на одне робоче місце, та вартості одного квадратного метра виробничої площі:

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

$$B_{y\partial} = R_{cn}^1 S_y C_{nl}, \quad (7.9)$$

де:  $R_{cn}^1$  – кількість робочих місць виконавців, шт. Приймаємо 8 робочих місць;

$S_y$  – питома площа на одне робоче місце,  $m^2$ ;

$C_{nl}$  – вартість одного квадратного метра площі, грн.

Згідно даних ТОВ науково-дослідницького консалтингового підприємства «Пектораль» (м. Кіровоград, вул. Глинки 16) ціна одного квадратного метра площі новобудови, вік якої не перевищує 25 років, по місту складає 800...1600 у.о./ $m^2$ . Враховуючи, що курс для розрахунку приймаємо 1 у.о. = 25 грн., вартість одного метра квадратного складе 20000 грн./ $m^2$ . На кожне робоче місце у середньому потрібно 8  $m^2$ . З урахуванням цього:

$$B_{y\partial} = 8 \cdot 8 \cdot 20000 = 1280000 \text{ грн.}$$

Вартість передавальних пристроїв складає 10% від вартості будівель, і у даному випадку вона складе: 128000 грн.

Балансова вартість інвентарю розраховується за нормою 3500 грн. на одне робоче місце. Тобто:

$$I_{nb} = R_{cn}^1 \cdot C_m, \quad (7.10)$$

де:  $C_m$  – ціна меблів для одного робочого місця, грн.

$$I_{nb} = 8 \cdot 3500 = 28000 \text{ грн.}$$

Балансова вартість обчислювальної техніки визначається по оптовим цінам постачальника з врахуванням витрат на транспортування.

Специфікація на обчислювальну техніку наведена в таблиці 7.7.

Дані по оптовій ціні на обладнання та комплектуючі вибирались по прайсу фірми Brain за 5.11.21 – джерело <http://brain.com.ua>.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

Таблиця 7.6 – Специфікація

Найменування комплектуючої або обладнання	Тип	Оптова ціна
Персональний комп'ютер		11186
Системний блок		6490
Процесор	Intel Core i5-750 S-1156 (8M Cache, 2.6 GHz)	1638
Системна плата	MB Asrock H55M-LE s1156 (H55, s1156) DDR3 2600(OC)x, ntel(R) HD Graphics 1xPCI-E 16x, 4xSATA2, Lan 1000 Mb/s, SB 7.1) mATX	1050
Відеокарта	VC Manli GeForce GT220 PCIe-2.0 512MB DDR2 128 bit HDMI DVI RTL	860
Жорсткий диск	HDD 640 Gb WD 7200 64Mb WD6400AAR SATAII	980
Оперативна пам'ять	DIMM 2048Mb DDR3 PC3-10600 Samsung 1333Mhz	468
DVD-привод	Super Multi LG SATA DVD±RW R+22x/22x, RW+8x/-6x, DL+16x/-12x, RAM 12x SecurDisc, black (GH22NS40RBB)	432
Корпус	ASUS TA-861 500W FSP ATX-500W (Black/Silver panel) ATX (90-PL861AF5C4 53CZ)	822
Кулер	–	–
Кардрідер внутрішній	USB 2.0 Card reader STORM CR-35U1A4-Black int. 3.5", 1*USB2.0+AUDIO+1394, multi: All Type Cards, black	240
інше	Клавіатура, мишка	-

Продовження таблиці 7.6

Найменування комплектуючої або обладнання	Тип	Оптова ціна
Монітор	22" TFT, ASUS VW223D ( 5ms, 300/3000: 170/160, D-SUB, Wide)	3200
Принтер лазерний	Canon i-SENSYS LBP6030W	2700
Принтер струминний	Epson Stylus Photo P50 (C11CA45341) + USB cable	5500
Сканер	Epson Perfection V37 Photo	2970
Копіювальний апарат	Canon i-SENSYS MF217W with Wi-Fi	5965
Пристрій безперебійного живлення	UPS APC BACK-UPS ES 525VA 230V (BE525-RS)	1496

Витрати на транспорт, монтаж та випробування можуть бути прийняті в межах до 10% від оптової ціни. Для визначення необхідної кількості капітальних вкладень складемо таблицю 7.8.

Таблиця 7.7 – Балансова вартість обчислювальної техніки

Найменування обчислювальної техніки	Кількість, шт.	Ціна за одиницю, грн.	Витрати на транспортування, монтаж та випробування.	Загальна вартість, грн.
Персональні комп'ютери	8	11186	8948,8	98436,8
Принтер лаз.	2	2700	540	5940
Принтер струм.	1	5500	550	6050
Сканери	1	2970	8948,8	98436,8
Копіюв. апарат	1	5965	540	5940
Всього	—	—	—	214803,6

Таблиця 7.8 – Вартість основних фондів та амортизаційні відрахування розробника

Групи та види основних фондів	Балансова вартість, грн.	Амортизація	
		Норма, %	Відрахування, грн.
1	2	3	4
Група 3			
1. Будівлі	1280000	-	-
2. Передавальні пристрої	128000	-	-
Всього по групі	1408000	5	70400
Група 4			
3. Обчислювальна техніка	214804	-	-
Всього по групі	214804	50	107402
Група 5, 6			
4. Вимірювальні пристрої	5190	25	-
5. Транспортні засоби	0	20	-
6. Господарський інвентар	28000	25	-
Всього по групі	33190	-	8297,5
Нематеріальні активи			
7. Нематеріальні активи	22000	10	2200
Разом	$K_p = 1677994$		$A_p = 188299,5$

Примітка: вартість автомобіля приймаємо рівною нулю.



Згідно виданих викладачем норм приймаємо 0,5 пачки паперу на місяць розробки. Тоді, враховуючи, що вартість пачки паперу складає  $C_n = 100$  грн., визначаємо вартість паперу за період розробки  $N_m = 3$  міс:

$$Z_{M1} = C_n \cdot N_m. \quad (7.16)$$

$$Z_{M1} = 100 \cdot 0,5 \cdot 3 = 150 \text{ грн.}$$

Згідно виданих викладачем норм до вартості запам'ятовуючих пристроїв входить вартість CD дисків в кількості, що дорівнює кількості екземплярів програм та одного DVD диска для збереження резервної копії програми:

$$Z_{M2} = \sum C_d, \quad (7.17)$$

де:  $C_d$  – вартість дисків CD/DVD: CDR TDK 700Mb, 80Min, 52x Cake box – 12 грн./шт., DVD-R LG 4,7Gb, 16x speed Cake box – 12 грн./шт.

$$Z_{M2} = 12 \cdot 22 + 12 = 276 \text{ грн.}$$

Згідно виданих викладачем норм одноразовій заправці підлягають усі друкуючі пристрої і становить:

$$Z_{M3} = \sum C_z, \quad (7.18)$$

де:  $C_z$  – вартість розхідних матеріалів друкуючих пристроїв: відновлення та заправка картриджу для Canon i-SENSYS LBP6030W – 574 грн.; картридж для Epson Stylus Photo P50 – 558 грн.; відновлення картриджу для MF217W – 570 грн.

$$Z_{M3} = 574 + 558 + 570 = 1702 \text{ грн.}$$

$$Z_M = (150 + 276 + 1702) / 22 = 97 \text{ грн.}$$

Визначимо витрати на освоєння нових мов програмування або операційних систем за нормативом ( $H_n = 15\%$ ) від основної зарплати виконавців:

$$O_n = Z_o \cdot H_n \cdot 0,01, \quad (7.19)$$

де:  $H_n$  – норматив витрат на освоєння нових мов програмування, %.

$$O_n = 5700 \cdot 15 \cdot 0,01 = 855 \text{ грн.}$$

Визначимо витрати на амортизацію основних фондів з урахуванням загальної річної суми амортизаційних відрахувань та кількості екземплярів програм ( $N_e = 22$  прим.):

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

$$A_m = \frac{A_p \cdot N_{\text{міс}}}{N_e \cdot 12}, \quad (7.20)$$

де:  $A_p$  – загальна річна сума амортизаційних відрахувань, грн.

$$A_m = 188300 \cdot 3 / (22 \cdot 12) = 2140 \text{ грн.}$$

Повна собівартість ПЗ визначається як сума витрат за попередніми статтями калькуляції:

$$C_n = Z_o + Z_d + C_{oc} + \Gamma_{ocn} + Z_m + O_n + A_m. \quad (7.21)$$

$$C_n = 5700 + 570 + 1379 + 855 + 97 + 855 + 2140 = 11596 \text{ грн.}$$

Визначимо плановий прибуток за рівнем рентабельності ( $P_n$ ) програмної продукції, яка залежить від складності програми та ступеня новизни задачі.

Для даного програмного забезпечення рівень рентабельності складає 50%.

$$P_p = 0,01 \cdot P_n \cdot C_n, \quad (7.22)$$

де:  $P_n$  – рівень рентабельності, %.

$$P_p = 0,01 \cdot 50 \cdot 11596 = 5798 \text{ грн.}$$

Величини ціна підприємства, податок на додану вартість, відпускна ціна програмної продукції визначаються за формулами, приведеними в таблиці 7.9

Таблиця 7.9 – Нормативна калькуляція собівартості розробки програмного забезпечення задачі

Найменування статей витрат	Позначення	Величина, грн
1	2	3
1. Основна зарплата виконавців	$Z_o$	5700
2. Додаткова зарплата виконавців	$Z_d$	570
3. Відрахування на соціальні потреби	$C_{oc}$	1379
4. Загальногосподарські витрати	$\Gamma_{ocn}$	855
5. Витрати на матеріали	$Z_m$	97
6. Освоєння нових операційних систем, мов програмування	$O_n$	855

Продовження таблиці 7.9

1	2	3
7. Амортизація основних фондів	$A_m$	2140
8. Повна собівартість програмного забезпечення	$C_n$	11596
9. Плановий прибуток	$P_p$	5798
10. Ціна підприємства $C_n = C_n + P_p$	$C_n$	17394
11. Податок на додану вартість $ПДВ = 0.01 \cdot H_{об} \cdot C_n$	$ПДВ$	3478,8
12. Відпускна ціна програмної продукції $C = C_n + ПДВ$	$C$	20872,8

**7.6 Визначення об'єму капітальних вкладень у споживача програмної продукції**

Об'єм капітальних вкладень у споживача програмної продукції визначаємо на основі балансової вартості основних фондів, яка враховує ціну, транспортно-заготівельні витрати, вартість будівель, монтажних та пусконаладжувальних робіт, а також витрати на випробування у виробничих умовах. Результати розрахунків зводимо у таблицю 7.10.

Таблиця 7.10 – Розрахунок об'єму капітальних вкладень у споживача програмної продукції

Найменування капітальних вкладень	Сума за варіантами, грн.	
	Базовий	Новий
Вартість програмної продукції	–	20873
Всього капітальних витрат	–	20873

## 7.7 Визначення експлуатаційних витрат

Експлуатаційні витрати у споживача програмної продукції визначаємо при умові роботи підсистеми на протязі року. Результати зводимо до таблиці 7.11.

Таблиця 7.11 – Розрахунок експлуатаційних витрат у споживача програмної продукції

Найменування статей витрат	Позначення	Сума витрат за варіантами, грн.	
		Базовий	Новий
1. Витрати на обслуговування	$Z_p$	40260	12078
2. Витрати на електроенергію	$Z_{ел}$	-	-
3. Витрати на амортизацію	$Z_{ам}$	-	5218
Всього витрат за рік	$I$	40260	17296

Витрати на обслуговування:

$$Z_p = T_p \cdot Z_z \cdot (1 + 0,01 \cdot H_q) \cdot (1 + 0,01 \cdot H_c), \quad (7.23)$$

де:  $T_p$  – кількість годин, що витрачається на обслуговування на рік, год.

(приймаємо 1000 год.);

$Z_z$  – заробітна плата обслуговуючого персоналу, грн/год.

$$Z_{p \text{ баз}} = 1000 \cdot 30 \cdot 1,1 \cdot 1,22 = 40260 \text{ грн},$$

до:

$$Z_{p \text{ нов}} = 300 \cdot 30 \cdot 1,1 \cdot 1,22 = 12078 \text{ грн}.$$

Витрати на електроенергію визначаються з урахуванням споживаємої потужності ( $P_{ел}$ ) в кіловатах, часу експлуатації технічних засобів ( $T_p$ ) в годинах та ціни однієї кіловат-години ( $C_{ел}$ ):

$$Z_{ел} = P_{ел} \cdot T_p \cdot C_{ел}. \quad (7.24)$$

$$Z_{ел \text{ баз}} = Z_{ел \text{ нов}}$$

Витрати по амортизації визначаються на основі норм амортизаційних відрахувань, вартості програмної продукції і основних фондів. Для розрахунку складаємо таблицю 7.12.

Таблиця 7.12 – Розрахунок амортизаційних відрахувань

Групи основних фондів	Норма амортизації %	Балансова вартість, грн., за варіантами		Сума відрахувань, грн за варіантами	
		Базовий	Новий	Базовий	Новий
Програмна продукція	25	–	20873	–	5218,25
Всього відрахувань	-	–	20873	–	5218,25

### 7.8 Визначення економічної ефективності програмної продукції

Економічна ефективність програмного забезпечення визначається для виготовлювача і споживача за такими показниками.

Величина економічного ефекту при виготовленні програмної продукції, розраховуємо за формулою:

$$E_e = (C_n - C_n) \cdot N_e - \sum_{i=1}^m E_{p_m} \cdot K_{p_m}, \quad (7.25)$$

де:  $K_p$  – балансова вартість основних фондів розробника, грн.;  $E_p$  – розрахунковий коефіцієнт капіталовкладень.

$$E_e = (17394 - 11596) \cdot 22 - (0,05 \cdot 1408000 + 0,5 \cdot 214804 + 0,25 \cdot 33190 + 0,1 \cdot 22000) \cdot 3/12 = 80481 \text{ грн.}$$

Визначимо період окупності додаткових капітальних вкладень у виробника програмної продукції:

$$T_e = \frac{K_p}{(C_n - C_n) \cdot N_e}, \quad (7.26)$$

де:  $K_p$  – балансова вартість основних фондів розробника.

$$T_e = \frac{1677994}{(17394 - 11596) \cdot 21 \cdot 12 / 3} = 3,3 \text{ років.}$$

Визначимо величину економічного ефекту у користувача програмної продукції за формулою:

$$E_{cn} = (I_б - I_n) - E_n(K_n - K_б), \quad (7.27)$$

де:  $I_б$ ,  $I_n$  – величина експлуатаційних витрат за базовим и новим варіантом відповідно;  $K_б$ ,  $K_n$  – об'єм капітальних вкладень за варіантами, що порівнюються.

$$E_{cn} = (40260 - 17296) - 0,25 \cdot 20873 = 17746 \text{ грн.}$$

Таблиця 7.13 – Показники економічної ефективності програмної продукції

Найменування показників	Одиниця виміру	Величина
1. Кількість екземплярів програми	Прим.	22
2. Повна собівартість розробленої програми	Грн.	11596
3. Ціна розробленої програми	Грн.	17394
4. Плановий прибуток від реалізації одної програми	Грн.	5798
5. Рентабельність програмної продукції	%	50
6. Об'єм додаткових капітальних вкладень у виробника програмної продукції	Грн.	1677994
7. Загальний прибуток від реалізації програмної продукції	Грн.	127556
8. Величина економічного ефекту при виготовлені програмної продукції	Грн.	80481
9. Період окупності додаткових капітальних вкладень у виробника програмної продукції	Років	3,3
10. Об'єм додаткових капітальних вкладень у споживача програмної продукції	Грн.	20873
11. Величина економічного ефекту у користувача програмної продукції	Грн.	17746
12. Період окупності додаткових капітальних вкладень у користувача програмної продукції	Років	0,9

Визначимо період окупності додаткових капітальних вкладень у споживача програмної продукції за рахунок зниження експлуатаційних витрат:

$$T_{cn} = \frac{K_n - K_b}{I_b - I_n}, \quad (7.28)$$

$$T_{cn} = \frac{20873}{40260 - 17296} = 0,9 \text{ року.}$$

Показники економічної ефективності програмної продукції зводимо до таблиці 7.13.

### 7.9 Висновки

Розроблена програма економічно вигідна. За рахунок впровадження програмного забезпечення досягається скорочення часу обробки інформації, підвищується культура праці, підвищення якості приймаючих управлінських рішень.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

## 8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

### 8.1 Вступ

Протягом усієї історії людство приділяє прискіпливу увагу безпеці життя. Охорона праці є складовою частиною безпеки життя [1].

Законом України “Про охорону праці” [2] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та ДСанПіН 3.3.2-007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин».

Програмісти у процесі роботи мають негативний вплив на органи зору, а також мають значну розумову напругою і нервово-емоційне навантаження. Руки (суглоби пальців та м'язи рук) при роботі з клавіатурою мають теж істотне навантаження. До шкідливих факторів, які впливають на робітників галузі інформаційних технологій (ІТ) спеціалісти відносять високочастотні електромагнітні коливання (випромінювання) роботи апаратної частини ЕОМ та виділення шкідливих газів.

Ці шкідливі фактори можуть привести до професійних захворювань.

При розгляді шкідливих чинників роботи програмістів та інших спеціалістів ІТ будемо керуватись наступними нормативно-правовими актами: «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98, та

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

«Правила охорони праці під час експлуатації електронно-обчислювальних машин» НПАОП 0.00-1.28-10,

Умови праці програміста вуючають наступні фактори:

- параметри повітряного середовища в приміщенні;
- вентиляція приміщення;
- освітлення приміщення;
- параметри повітряного середовища в приміщенні, тощо.

Щоб запропонувати заходи щодо зменшення впливу комп'ютера на організм програміста визначемо фактори, які можуть викликати професійне захворювання і впливають на працездатність програміста,

## 8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером

Програміст працює з електронно-обчислювальною машиною (ЕОМ) та іншим обладнанням, яке є джерелом небезпеки ураження електричним струмом. Так як робота програміста характеризується істотним зоровим навантаженням, то вимагає належного освітлення. Так як програміст постійно перебуває в приміщенні, тому для комфортних умов праці в цьому приміщенні необхідно створити належний мікроклімат.

При роботі з використанням ЕОМ відзначають наступні небезпечні та шкідливі фактори:

- ризик виникнення надзвичайних ситуацій природного або штучного характеру на об'єкті або території.
- ризик виникнення пожежі;
- негативний вплив на органи зору людини;
- ризики ураження електричним струмом;
- недостатня, або надмірна освітленість робочого місця;
- монотонність праці;
- електромагнітні (у т.ч. високочастотні) випромінювання (коливання);

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

- несприятливі мікрокліматичні умови;
- нервово-емоційна напруженість праці;
- інтелектуальні навантаження;
- невідповідність ергономічних показників робочого місця діючим вимогам;
- шуми;
- статичні навантаження на кістково-м'язовий апарат;

### 8.3 Аналіз санітарно-гігієнічних умов праці на робочому місці програміста

Розглянемо умови праці у приміщенні, в якому працюють програмісти. Геометричні розміри приміщення наведено у таблиці 8.1.

Таблиця 8.1 – Розміри приміщення

Найменування	Значення, м
Ширина	6,6
Довжина	7,2
Висота	3

Таблиця 8.2 – Площа та обсяг приміщення, на одного працюючого

Геометрична характеристика	Одиниця виміру	Нормативне значення*	Фактичне значення
Площа, S	м <sup>2</sup>	не менше 6.0	6,8
Обсяг, V	м <sup>3</sup>	не менше 20.0	20,4

\* Згідно ДСанПіН 3.3.2.007-98 (Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин).

У зазначеному приміщенні працює 7 осіб. За даними, які наведено у табл. 8.1 та табл. 8.2, можна зробити висновок, що площа та об'єм приміщення у розрахунку на одно робоче місце програміста відповідають нормативним вимогам (Наказу Міністерства соціальної політики України № 207, від 14.02.2018 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», ДСанПіН 3.3.2-007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» та НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин»).

Температура повітря в приміщенні визначається впливом температури зовнішнього повітря і тепловою енергією, яка виділяється всередині приміщення. Джерелами виділення теплоти в даному приміщенні є електроустаткування, освітлювальні прилади, а також люди. У світлий час доби джерелом надлишкового тепла є сонячна радіація. Згідно Постанови № 42 від 01.12.1999 Головного державного санітарного лікаря України, робота, яка виконується в даному приміщенні, відноситься до категорії Іа. В цьому випадку людина витрачає енергії до 120 ккал у годину. Вологість повітря у приміщенні визначається впливом багатьох факторів, серед яких: вологість атмосферного повітря, виділення вологи людьми (при диханні та випарами з поверхні шкіри).

Мікроклімат повітряного середовища в приміщенні характеризується запиленістю та загазованістю повітря. Мікроклімат приміщення визначається діючим на організм людини поєднанням, вологості, температури, швидкості руху повітря та інтенсивності теплового випромінювання. Аналіз мікроклімату складається з визначення зазначених вище факторів і порівняння результатів із встановленими нормами.

У таблиці 8.3 наведено оптимальні та фактичні значення параметрів мікроклімату як для категорії ваги робіт Іа, так і розглянутого приміщення. У

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

приміщеннях, де встановлено ЕОМ, рекомендується застосування тільки оптимальних значень показників мікроклімату.

Таблиця 8.3 – Оптимальні і фактичні значення параметрів мікроклімату

Пора року	Оптимальні для Іа			Фактичні		
	Температура, °С	Воло- гість,%	Швидкість повітря, м/с	Температура, °С	Воло- гість%	Швидкість повітря, м/с
Холодна	22-24	40-60	0,1	22-23	41-58	0,12
Тепла	23-25	50-70	0,1	24-25	52-68	0,11

Проведений аналіз показує, що показники мікроклімату в приміщенні відповідають установленим нормам. Штучне опалення застосовується у холодний період року.

В літню пору застосовується кондиціонер.

Для боротьби з пилом робляться регулярні провітрювання та вологі прибирання приміщенні.

У приміщенні знаходяться наступні джерела шуму: принтер *Canon PIXMA G1411*, електродвигуни вентиляторів ЕОМ.

Одним з найважливіших факторів, які впливають на ефективність трудової діяльності людини, та попереджають травматизм і професійні захворювання програмістів є освітлення на робочому місці.

Працю працівника, який постійно працює за комп'ютером, згідно ДБН В.2.5 – 28 – 2006 р. можна віднести до роботи з малою точністю (найменший розмір об'єкта розрізнення від 1 до 5 мм) V-го розряду зорової роботи, з великою контрастністю об'єкта розрізнення (символів на екрані дисплея), з темним тлом (під розряд зорової роботи В). Приміщення можна віднести до 1-ої групи приміщень, у яких проводиться розрізнення об'єктів зорової роботи при фіксованому напрямку лінії зору того, що працює на робочу поверхню. Для такого типу приміщень і розряду зорової роботи нормоване значення коефіцієнта природної освітленості (КПО) робочої поверхні (при поєднаному, спільному

освітленні), повинен становити не більше 1,5%, освітленість при штучному висвітленні повинна становити 300 лк. Крім того все поле зору повинне бути освітлено достатньо рівномірно – ця основна гігієнічна вимога. Так як яскраве світло на ділянці периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими.

#### **8.4 Розробка заходів з умов поліпшення охорони праці**

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язково наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга).

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

## 8.5 Розрахункова частина

Початкові дані для розрахунку захисного заземлення:

Тип заземлення: робоче заземлення нульової точки трансформатора. Заземленню підлягає виробниче обладнання організації. Напруга – 220/380 В. Розташування заземлюючих електродів – по контуру.

Розрахунок проводиться за допустимим опором розтіканню струму заземлювача методом коефіцієнта використання заземлювачів.

Початкові дані для розрахунку захисного заземлення: тип верхнього шару ґрунта – супісок, нижнього шару ґрунта – чернозем. Умовна товщина верхнього шару ґрунта:  $H=1$  м. Для захисного заземлення: застосовуються вертикальні електроди – прутки довжиною  $L = 2$  м. Відстань між вертикальними заземлювачами (електродами)  $A=2$  м. Діаметр вертикального електрода (прутка)  $D=40$  мм, Тип горизонтального заземлювача: металева полоса. Розміри перетину з'єднуючої полоси: 100 x 8 мм. ( $b=100$  мм.). Опір заземлювача, який нормується:  $R_{3H} = 4$  Ом. Глибина закладення контура заземлення  $t=0,7$  м.

Розрахунок захисного заземлення можна автоматизувати за допомогою програми, сирцевий код якої опублікован на стр.13-16 [4]. Додаткову перевірку отриманих результатів можна робити за допомогою он-лайн сервісу URL: <https://kalk.pro/electricity/earthing/>.

Розрахунок.

Відстань від центра вертикального заземлювача до поверхні землі:

$$T=t+L/2=0,7+2/2=1,7 \text{ м.}$$

Еквівалентний питомий опір ґрунта [3]:

$$\rho_{\text{екв}} = \psi \rho_1 \rho_2 L / [\rho_1 \psi(L-H+t) + \rho_2 \psi(H-t)] = 194,5 \text{ Ом}$$

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

де

$\psi = 1,1$  табличне значення коефіцієнта сезонності у багатошаровому ґрунті;

$\rho_1 = 150 \text{ Ом}\cdot\text{м.}$  – табличне значення питомого опору верхнього шару ґрунта;

$\rho_2 = 100 \text{ Ом}\cdot\text{м.}$  – табличне значення питомого опору нижнього шару ґрунта.

Опір розтіканню електричного струму одного електрода вертикального заземлювача:

$$R_0 = [\rho_1 / (2\pi * L)] * [(ln) 2L/D) + 0,5 ln[(4T+L)/(4T-L)] = 75,97 \text{ Ом.}$$

Опір розтіканню електричного струму заземлювача, який нормується, якщо  $\rho_{\text{екв}} > 100$  (у нас  $\rho_{\text{екв}} = 194,5 > 100$ ):

$$R = R_{3H} * \rho_{\text{екв}} / 100 = 4 * 194,5 / 100 = 7,76 \text{ Ом.}$$

Опір розтіканню електричного струму горизонтального заземлювача (полоси):

$$R_{II} = 0,366 (\rho_{\text{екв}} \psi / L_{II} \cdot \eta_{II}) \lg(2 \cdot L_{II} * L_{II} / bt) = 7,9 \text{ Ом.}$$

де  $\eta_{II} = 0,27$  – табличне значення коефіцієнта використання горизонтального заземлювача, залежить від розташування (в ряд або по контуру) та співвідношення  $A/L$ ; довжина горизонтального заземлювача (полоси) при розташуванні заземлювачів по контуру:

$$L_{II} = A * n = 2 * 20 = 40 \text{ м.};$$

де  $n$  – ітераційна кількість вертикальних заземлювачів.

Загальний опір штучного заземлюючого пристрою розтіканню електричного струму на землю:

$$R_B = R_{II} R / (R_{II} - R) = 3,99 \text{ Ом.}$$

Кількість вертикальних заземлювачів:

$$n = R_0 / R_B * \eta_B = 20 \text{ шт.}$$

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

де  $\eta_B = 0,47$  – табличне значення коефіцієнта використання вертикального заземлювача, залежить від розташування (в ряд або по контуру) та співвідношення  $A/L$ .

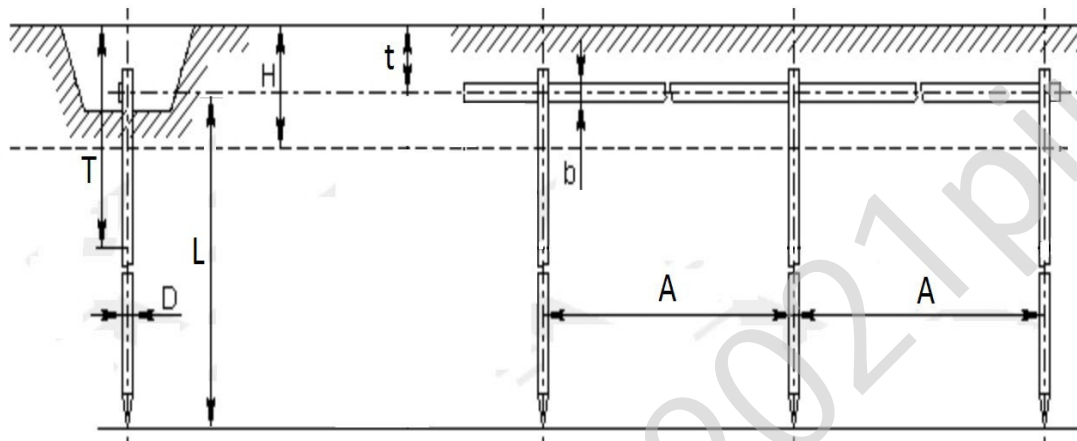


Рисунок 8.1 – Схема штучного заземлювача

## 8.6 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз приміщення, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з охорони праці.

Вим.	Арк.	№ докум.	Підпис	Дата

ВКРМ-123.21.0022.00.00.ПЗ

Арк.

73

## 9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи кібербезпеки для виявлення вразливостей web-застосунків.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів кібербезпеки для виявлення вразливостей web-застосунків.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем кібербезпеки для виявлення вразливостей web-застосунків.

– Досліджена система кібербезпеки для виявлення вразливостей web-застосунків.

– На основі отриманих результатів досліджень створена програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання кібербезпеки для виявлення вразливостей web-застосунків.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		74

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Java. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows XP/Vista/7/8/10.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Blowfish.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Розроблена програма має реальний економічний ефект від її впровадження у виробництво у сумі 17746 грн. З урахуванням вартості розробки програми та обладнання, строк окуплення становить 0,9 роки.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Толмачов Ю.П. Дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків // Збірник праць молодих науковців ЦНТУ. – Вип. 12. – Кропивницький: ЦНТУ, 2022.
2. Смірнов О.А./ Методичні рекомендації до виконання й захисту кваліфікаційної магістерської роботи/ Смірнов О.А., Гермак В.С., Мелешко Є.В., Коваленко О.В., Якименко Н.М., Буравченко К.О./ Кропивницький – 2019, 55 с.
3. Різновиди веб-систем та їх основні вразливості / Електронний ресурс/  
<http://websecurity.com.ua/security>
4. OWASP. Стандарт оцінювання відповідності безпеки додатків 3.0 OWASP – 2015. – 73 с.
5. OWASP/ Десять найбільш критичних ризиків для безпеки веб-додатків – 2013. – 22 с.
6. 5. Penetration Test Guidance Special Interest Group PCI Security Standards Council. PCI Data Security Standard (PCI DSS) – 2015. – 43 с.
7. Фреймворки в веб-розробке [Електронний ресурс] // Web Creator. – 2015. – Режим доступу до ресурсу: [https://webcreator.ru/articles/about\\_frameworks](https://webcreator.ru/articles/about_frameworks).
8. Захист веб-додатків [Електронний ресурс]. – 2010. – Режим доступу до ресурсу: [http://www.ereading.club/bookreader.php/1012355/DJ-Andrey-sXe\\_-\\_Zaschita\\_veb-prilozheniy.html](http://www.ereading.club/bookreader.php/1012355/DJ-Andrey-sXe_-_Zaschita_veb-prilozheniy.html).
9. Євтеєв Д. Методи обходу Web Application Firewall [Електронний ресурс] / Дмитро Євтеєв. – Режим доступу до ресурсу: <http://www.ptsecurity.ru/download/PTdevteev-CC-WAF.pdf>.
10. Основні методи тестування безпеки веб-додатків [Електронний ресурс]. – 2013. – Режим доступу до ресурсу: <http://uastudent.com/osnovni-metody-testuvannja-bezpekyveb-dodatktiv/>.
11. Седерхольм Д. Пуленепробиваемый Web-дизайн.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

12. Повышение гибкости сайта и защита от потенциальных неприятностей с помощью XHTML и CSS / Ден Седерхольм, 2006. – 256 с. – (Школа Web-мастерства).

13. Жуков Ю.В. Основы веб-хакинга. Нападение и защита / Юрий Викторович Жуков, 2012. – 206 с.

14. 2. MVC – модель-представление-контроллер [Электронный ресурс] // WebCreator. – 2014. – Режим доступа до ресурсу: <https://webcreator.ru/articles/mvc>.

15. 3. Information technology – Vocabulary, 2015. – (ISO/IEC 2382:2015)

16. 97% of Java Apps Harbor a Known Security Hole [Электронный ресурс] //Fortune.com. – 2016. – Режим доступа до ресурсу:

17. State of app security 2016: Most common vulnerabilities, top trends [Электронный ресурс] // TechBeacon.com. – 2016. – Режим доступа до ресурсу: <https://techbeacon.com/security/state-app-security-2016-most-common-vulnerabilitiestop-trends>.

18. Microsoft SDL [Электронный ресурс] – Режим доступа до ресурсу: <https://www.microsoft.com/en-us/securityengineering/sdl>.

19. OWASP Secure Coding Practices Quick Reference Guide. // OWASP. – 2010.

20. Report: Cenzic Application Vulnerability Trends Report: 2014 [Электронный ресурс] // Cenzic. – 2014. – Режим доступа до ресурсу: <https://info.cenzic.com/2013-Application-Security-Trends-Report.html>.

21. 9. OWASP Top 10 – 2017 The Ten Most Critical Web Application Security Risks, 2017. – (creative commons)

22. Дональд Кнут/ Искусство программирования, том 1. Основные алгоритмы/ Дональд Кнут/ – 3-е изд. – М.: «Вильямс», 2006. – С. 720. – ISBN 0-201-89683-4

23. Юрченко І.В./ Інформатика та програмування. Частина 1. Навчальний посібник./ Юрченко І.В – Чернівці: Книги–ХХІ, 2011.– 203 с.

					ВКРМ-123.21.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

24. Юрченко І.В./ Інформатика та програмування. Частина 2/ Юрченко І.В., Сікора В.С.– Чернівці: Видавець Яворський С.Н., 2015.– 210 с.
25. Семенюк А.Д./ Програмування. Практикум / Укл.: Семенюк А.Д., Сопронюк Ф.О. – Чернівці: Рута, 2010.– 143 с.
26. Фаронов В.В./Delphi. Программирование на языке высокого уровня./ Фаронов В.В. – СПб.: Питер, 2004.– 640 с.
27. Гофман В.Э./ Delphi./ Гофман В.Э., Хомоненко А.Д – М., 2012.– 1145 с.
28. Культин Н.Б./ Delphi. Программирование на Object Pascal/ Культин Н.Б.– М., 2012.– 526 с.
29. Комп'ютерні мережі: Методичні вказівки до лабораторних робіт / Укл.: В.С. Сікора, І.В. Юрченко.– Чернівці: Рута, 2002.– 43 с.
30. В.С. Сікора/ Операційна система Microsoft Windows: / Укл.: В.С. Сікора, І.В. Юрченко.– Чернівці: Рута, 2013.– 48 с.
31. В. Ф. Шаньгіна / Інформаційна безпека комп'ютерних систем і мереж: Навчальний посібник / В. Ф. Шаньгіна. – М .: ИД ФОРУМ: ИНФРА -М, 2012. – 416 с.
32. Гуда, А. Н./ Інформатика і програмування. Комп'ютерний практикум: навч. посібник для вузів / Гуда, А. Н., Бутаков, М. А., Нечитайло, Н. М., Чернов, А. В. – М .: Дашков і К, 2010. – 240
33. Іванова, Г. С. / Технологія програмування. – М .: КноРус, 2011. – 336
34. Іртегов Д. В./ Введення в операційні системи: навч. посібник.- СПб .: БХВ-Петербург, 2012.-1040 с.
35. Н. З. Ємельянова / Проектування інформаційних систем: навч. посібник / Н. З. Ємельянова, Т. Л. Партика, І. І. Попов. – М .: Форум, 2009. – 432 с.
36. Назаров С. В. /Операційні системи. Практикум: навч. посібник./ Назаров С. В. – М. КноРус, 2012-376 с.
37. Кузин А. В. / Комп'ютерні мережі: Навчальний посібник

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

[Електронний ресурс] / А. в. Кузін. – 3-е изд., Перераб. і доп. – М.: Форум: ИНФРА-М, 2011. – 192 с. – Режим доступу: <http://znanium.com/bookread.php?book=249563>

38. Корольов, А. Л. Комп'ютерне моделювання. Лабораторний практикум [Електронний ресурс] / А. Л. Корольов. – М.: БИНОМ. Лабораторія знань, 2012. – 296 с. : Ил. – (Педагогічну освіту). – ISBN 978-5-9963-0270-3.

39. Баранова, Е. К. Основи інформатики та захисту інформації [Електронний ресурс]: Навч. посібник / Є. К. Баранова. – М.: РІОР: ИНФРА-М, 2013. – 183 с.

40. Гагаріна Л. Р./ Технологія розробки програмного забезпечення./ Гагаріна Л. Р, Кокорева Е. В., Віснадул Б. Д. – М.: ІД. «Форум»: ИНФРА -М, 2011. – 400 с.

41. Голіцина О. Л. / Програмування на мовах високого рівня. Голіцина О. Л., Попов И. І / М.: ФОРУМ, 2011. – 496 с

42. Покришень Д.А. / Технологічні принципи проектування інформаційних систем / Д.А. Покришень // Сборник научных трудов SWorld. – Выпуск 1. Том 17. – Иваново: МАРКОВА АД, 2014. – С. 37-44

43. Саммервил И. / Инженерия программного обеспечения / И. Саммервил. – М.: Вильяме, 2002. – 720 с.

44. Смирнов А.А. Анализ и сравнительное исследование перспективных направлений развития цифровых телекоммуникационных систем и сетей / А.А.Смирнов, В.В.Босько, Е.В.Мелешко // Системи обробки інформації. – Х.: ХУ ПС, 2008. – Вип.7(74). – С.120-123.

45. Смирнов А.А. Усовершенствование метода управления очередями в многопротокольных узлах телекоммуникационной сети / А.А.Смирнов, Е.В.Мелешко // Збірник тез та доповідей другої всеукраїнської науково-практичної конференції «Системний аналіз. Інформатика. Управління». Запоріжжя. Тези доповідей. Запоріжжя: КПУ, 2011.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		79

46. Смирнов С.А. Метод безопасной маршрутизации метаданных в облачные антивирусные системы / А.К. Дидык, С.А. Смирнов // Информационные технологии в управлении, образовании, науке и промышленности: монография / Под редакцией профессора В.С. Пономаренко. – Х.: Видавець Рожко С.Г., 2016. – 566 с.

47. Смирнов С. А. Сравнительные исследования математических моделей технологии распространения компьютерных вирусов в информационно-телекоммуникационных сетях / Мохамад Абу Таам Гани, А. А. Смирнов, А. В. Коваленко, С. А. Смирнов // Системи обробки інформації: зб. наук. праць. – Х.: ХУПС, 2014. – Вип. 9(125). – 105-110.

48. Смирнов С. А. Математическая модель интеллектуального узла коммутации с обслуживанием информационных пакетов различного приоритета / Мохамад Абу Таам Гани, А. А. Смирнов, Н. С. Якименко, С. А. Смирнов // Збірник наукових праць Харківського університету Повітряних Сил. – Харків: ХУПС, 2014. – Вип. 4 (41). – С. 48-52.

49. Смирнов С. А. Исследование показателей качества функционирования интеллектуальных узлов коммутации в телекоммуникационных системах и сетях / Мохамад Абу Таам Гани, А. А. Смирнов, Н. С. Якименко, С. А. Смирнов // Наука і техніка Повітряних Сил Збройних Сил України: наук. журн. –Х.: ХУПС, 2014. – № 4(17). – С. 90-95.

50. Смирнов С. А. Усовершенствованный алгоритм управления доступом к «облачным» телекоммуникационным ресурсам / Мохамад Абу Таам Гани, А. А. Смирнов, Н. С. Якименко, С. А. Смирнов // Системи обробки інформації: зб. наук. праць. – Х.: ХУПС, 2015. –Вип. 1(126). – С. 150-153.

51. Smirnov S.A. Method of controlling access to intellectual switching nodes of telecommunication networks and systems / A.A. Smirnov, Mohamad Abou Taam, S.A. Smirnov // International Journal of Computational Engineering Research (IJCER). – Volume 5, Issue 5. – India. Delhi. – 2015. – P. 1-7.

52. Смирнов С. А. Анализ и исследование методов управления сетевыми ресурсами для обеспечения антивирусной защиты данных / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Системи озброєння і військова техніка: наук. журн. – Х.: ХУПС, 2015. – № 3(43). – С. 100-107.

53. Смирнов С. А. Исследование эффективности метода управления доступом к облачным антивирусным телекоммуникационным ресурсам / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Наука і техніка Повітряних Сил Збройних Сил України: наук. журн. – Х.: ХУПС, 2015. – № 3(20). – С. 134-141.

54. Смирнов С. А. Комплекс gert-моделей технологии облачной антивирусной защиты телекоммуникационной системы / А. А. Смирнов, А. К. Дидык, А. Н. Дреев, С. А. Смирнов // Безпека інформації: наук. – практ. журн. – К.: НАУ, 2015. – Т. 21, № 3. – С. 251-262.

55. Смирнов С. А. Метод безопасной маршрутизации метаданных в облачные антивирусные системы / А. А. Смирнов, А. К. Дидык, С. А. Смирнов // Системи озброєння і військова техніка: наук. журн. – Х.: ХУПС, 2016. – № 2 (46). – С. 146-149.

56. Смирнов С. А. Модели системы нейросетевых экспертов безопасной маршрутизации в облачных антивирусных системах / А. А. Смирнов, А. К. Дидык, А. Н. Дреев, С. А. Смирнов // Системи обробки інформації: зб. наук. праць. – Х.: ХУПС, 2016. – Вип. 3 (140). – С. 36-39.

57. Смирнов С. А. Метод безопасной маршрутизации на базовом множестве путей передачи метаданных в облачные антивирусные системы / В. Л. Бурячок, С. А. Смирнов // Системи управління, навігації та зв'язку. – Полтава, 2016. – Вип. 4(40). – С. 57-62.

58. Смирнов С. А. Способ контроля линий связи телекоммуникационной системы облачного антивируса / А. А. Смирнов, А. К. Дидык, А. Н. Дреев, С. А. Смирнов // Збірник наукових праць Харківського університету Повітряних Сил. – Харків: ХУПС, 2016. – № 2 (47). – С. 148-152.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		81

59. Смирнов С. А. Дослідження та реалізація GERT-моделі технології розповсюдження комп'ютерних вірусів для захисту телекомунікаційних систем / В. Л. Бурячок, Мохамад Абу Таам Гани, С. А. Смирнов // Інформаційні технології та комп'ютерна інженерія: зб. тез доп. наук.-практ. конф., м. Кіровоград, 4 грудня 2014 р. – Кіровоград: КНТУ, 2014. – С. 168.

60. Смирнов С. А. Исследование математических моделей технологии распространения компьютерных вирусов / А. А. Смирнов, Мохамад Абу Таам Гани, С. А. Смирнов // Актуальні питання забезпечення кібернетичної безпеки та захисту інформації: зб. наук. праць міжнар. наук.-практ. конф., м. Київ, 25-28 лютого 2015 р. – К.: Європейський університет, 2015. – С. 90-91.

61. Смирнов С. А. Метод управления доступом к «облачным» ресурсам для защиты телекоммуникационных систем / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Всеукраїнська науково-практична конференція «Інформаційна безпека держави, суспільства та особистості», м. Кіровоград, 16 квітня 2015 р.: зб. тез доп. – Кіровоград: КНТУ, 2015. – С. 50-52.

62. Смирнов С. А. Разработка метода управления доступом в интеллектуальных узлах коммутации / А. А. Смирнов, Мохамад Абу Таам Гани, С. А. Смирнов // Проблеми і перспективи розвитку ІТ-індустрії: зб. тез VII міжнар. наук.-практ. конф., м. Харків, 17-18 квітня 2015 р. – Х.: ХНЕУ, 2015. – С. 14.

63. Смирнов С. А. Реализация метода управления доступом в интеллектуальных узлах коммутации / А. А. Смирнов, Мохамад Абу Таам Гани, С. А. Смирнов // Збірник тез XVII міжнародного науково-практичного семінару «Комбінаторні конфігурації та їх застосування». м. Кіровоград. 17-18 квітня 2015 р. – Кіровоград: КНТУ. – 2015. – С. 91-92.

64. Смирнов С. А. технология передачи сигнатур в облачные антивирусные системы для обеспечения защищенности телекоммуникационных сетей / А. А. Смирнов, С. А. Смирнов // Збірник тез V міжнародної науково-

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82

технічної конференції «ITSEC», Київ, 19-22 травня 2015 р. – К.: НАУ 2015. – С. 12-13.

65. Смирнов С. А. Реализация математической модели интеллектуального узла коммутации для обеспечения защищенности телекоммуникационной сети / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Інформаційна та економічна безпека (INFECO-2015): зб. тез II Міжнар. наук.-практ. Інтернет-конф., м. Харків, 21-22 травня 2015 р. – Х.: ХІБС УБС НБУ, 2015. – С. 20-24.

66. Смирнов С. А. Разработка математической модели технологии распространения компьютерных вирусов в информационно-телекоммуникационных сетях / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Сборник тезисов XI международной конференции «Стратегия качества в промышленности и образовании», г. Варна, Болгария, 01-06 июня 2015 г. – Варна: ТУВ, 2015. – С. 488-491.

67. Смирнов С. А. Метод управления доступом к облачным телекоммуникационным ресурсам для обеспечения защиты данных / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Комп'ютерні технології та інформаційна безпека: зб. тез доп. міжнар. наук.-практ. конф., м. Кіровоград, 2-3 липня 2015 р. – Кіровоград: КНТУ, 2015. – С. 4-5.

68. Смирнов С. А. Имитационная модель системы управления доступом к облачным антивирусным телекоммуникационным ресурсам / Мохамад Абу Таам Гани, А. А. Смирнов, С. А. Смирнов // Збірник тез першої всеукраїнської науково-практичної конференції «Перспективні напрями захисту інформації» (м. Затока, 7-9 вересня 2015 р.). – Одеса: ОНАЗ, 2015. – С. 90-94.

69. Смирнов С. А. Разработка комплекса gert-моделей технологии облачной антивирусной защиты телекоммуникационной системы / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Інформаційні технології та взаємодії» (IT & I): зб. тез II міжнар. наук. -практ. конф., м. Київ, 3-5 листопада 2015 р. – К.: КНУ ім. Тараса Шевченка, 2015. – С. 65-67.

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

70. Смирнов С. А. Разработка моделей телекоммуникационной системы формирования и обработки метаданных в облачных антивирусных системах / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Информационные и телекоммуникационные технологии: образование, наука, практика: сб. тезисов II междунар. научно-практ. конф., г. Алматы, Казахстан, 3-4 декабря 2015 г. – Алматы: КазНИТУ им. К.И. Сатпаева, 2015. – С. 309-313.

71. Смирнов С. А. gert-модели технологии облачной антивирусной защиты / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Безпека українського суспільства в концепції вступу в постіндустріальне суспільство ЄС: зб. тез Круглого столу, м. Київ, 16 грудня 2015 р. – К.: Європейський університет, 2015. – С.41-43.

72. Смирнов С. А. Алгоритмы формирования множества маршрутов передачи метаданных в облачные антивирусные системы / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Актуальні питання забезпечення кібернетичної безпеки та захисту інформації: зб. наук. праць II Міжнар. наук.-практ. конф., м. Київ, 24-27 лютого 2016 р. – К.: Європейський університет, 2016. – С. 140-142.

73. Смирнов С. А. Разработка и реализация метода безопасной маршрутизации метаданных в облачные антивирусные системы / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Securitea informationala 2015-2016: Conferenta internationala (editia a XII-a), Chisinau, Moldova, 3 martie 2016. – Chisinau: ADSEM, 2016. – С. 90-96.

74. Смирнов С. А. Алгоритм формирования базового множества маршрутов передачи метаданных в облачные антивирусные системы / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Інформатика та системні науки (ІСН-2016): зб. тез VII всеукр. наук.-практ. конф., м. Полтава, 10-12 березня 2016 р. – Полтава: ПУЕТ, 2016. – С. 261-263.

75. Смирнов С. А. Система обработки и формирования начального состояния маршрутизации метаданных в облачные антивирусные системы /

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Проблемы кибербезопасности информационно-телекоммуникационных систем: сб. тез наук.-практ. конф., м. Київ, 10-11 березня 2016 р. – К.: КНУ ім. Тараса Шевченка, 2016. – С. 81-82.

76. Смирнов С. А. Алгоритм безопасной маршрутизации на базовом множестве путей передачи метаданных в программный сервер облачной антивирусной системы / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Інформаційна безпека та комп'ютерні технології (IS&CT): зб. тез міжнар. наук.-практ. конф., м. Кіровоград, 24-25 березня 2016 р. – Кіровоград: КНТУ, 2016. – С. 73.

77. Смирнов С. А. Исследование способа контроля линий связи телекоммуникационной системы для облачных антивирусов / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Збірник тез першої міжнародної науково-практичної конференції «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі» (ПНПЗК-2016), м. Харків, 30 березня – 1 квітня 2016 р. – Х.: НТУ «ХП», 2016. – С. 14.

78. Смирнов С. А. Разработка способа контроля линий связи телекоммуникационной системы для облачных антивирусов / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Матеріали XVIII міжнародного науково-практичного семінару «Комбінаторні конфігурації та їх застосування» (м. Кіровоград, 15-16 квітня 2016 р.). –Кіровоград: КНТУ, 2016. – С. 182-186.

79. Смирнов С. А. Разработка и исследование способа контроля линий связи телекоммуникационных сетей для облачных антивирусных систем / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Проблеми і перспективи розвитку ІТ-індустрії: VIII міжнар. наук.-практ. конф., м. Харків, 28-29 квітня 2016 р.: зб. тез. – Х.: ХНЕУ, 2016. – С. 48.

80. Смирнов С. А. Модель системы нейросетевых экспертов безопасной маршрутизации для облачных антивирусных систем / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Інформаційна та економічна безпека (INFECO-

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		85

2016): зб. тез III міжнар. наук.-практ. конф., м. Харків, 28-30 кві. 2016 р. – Х.: ХННІ ДВНЗ «УБС», 2016. – С. 178-182.

81. Смирнов С. А. Метод безопасной маршрутизации метаданных в облачные антивирусные системы / А. А. Смирнов, С. А. Смирнов, А. К. Дидык // Сборник тезисов XII международной конференции «Стратегия качества в промышленности и образовании» (г. Варна, Болгария, 30 мая – 02 июня 2016 г.). – Варна: ТУВ, 2016. – С. 581-585.

82. Смирнов С. А. Оценка эффективности метода безопасной маршрутизации метаданных в облачные антивирусные системы / А. А. Смирнов, С. А. Смирнов, А. С. Коваленко // РадіоЕлектроніка та ІнфоКомунікації: зб. тез першої наук. – техн. конф., м. Київ, 11-16 вересня 2016 р. – К.: НТУУ «КПІ», 2016. – С. 17.

83. Зеркалов Д. В. Охорона праці в Галузі: Загальні вимоги: навч. посіб. Київ: Основа. 2011. 551 с.

84. Про охорону праці: Закон України від 14.10.1992 р. № 2694-XII. Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/2694-12#Text>

85. Сакулин В.П., Шептовицкий В.М. Безопасность труда при монтаже и эксплуатации электроустановок / В.П.Сакулин, В.М.Шептовицкий. – Л. : “Колос”, 1973. – 238 с.

86. Охорона праці. Ч. 1. Захисне заземлення: метод. вказ. до викон. розрахунків з викор. персон. ЕОМ IBM сумісного типу / Кіровоград. ін -т с.-г. машинобуд.; [укл. О. В. Оришака, Є. К. Солових, В. О. Оришака]. – Кіровоград: КІСМ, 1997. – 20 с. Режим доступу до ресурсу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/4358>

					<b>ВКРМ-123.21.0022.00.00.ПЗ</b>	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		86

Додаток А  
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					<b>ВКРМ-123.21.0022.00.00.ТЗ</b>		
Вим.	Арк.	№ документа	Підпис	Дата			
Розробив	Толмачов Ю.П.				Літ.	Аркуш	Аркушів
Перевірів	Коваленко О.В.						
Н. Контр.	Гермак В.С.				ЦНТУ КІ-20М-1,4		
Затв.	Смірнов О.А.						
<i>Дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web- застосунків</i>					М	1	6

## **1 Найменування та область застосування**

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи кібербезпеки для виявлення вразливостей web-застосунків.

## **2 Підстава для розробки**

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 42-13 від 02.08.2021 року).

## **3 Мета та призначення розробки**

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація системи кібербезпеки для виявлення вразливостей web-застосунків.

## **4 Джерела розробки**

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

## **5 Технічні вимоги**

### **5.1 Склад продукції**

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;

					<b>ВКРМ-123.21.0022.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

## 5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи кібербезпеки для виявлення вразливостей web-застосунків;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

## 5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

## 5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

## 5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					<b>ВКРМ-123.21.0022.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

## 5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

## 5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ архітектури IBM PC, працювати в ОС Windows XP/Vista/7/8/10 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

## 5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows XP/Vista/7/8/10.

### 5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

### 5.8.2 Мова програмування

Середовище Java.

					ВКРМ-123.21.0022.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

### 5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

### 5.8.4 Вихідні дані

Робоча програма.

## 6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

## 7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати розрахунок витрат показників економічного ефекту з урахуванням цін на 3 вересня 2021 року.

## 8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинні бути розглянуті шкідливі і небезпечні фактори при роботі з комп'ютером.

					ВКРМ-123.21.0022.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

## 9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 129 аркушів.

## 10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Розрахунок з техніко-економічного обґрунтування.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

## 11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 10.12.2021 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 20.12.2021 р.

					<b>ВКРМ-123.21.0022.00.00.ТЗ</b>	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б  
(обов'язковий)

**Міністерство освіти і науки України**  
**Центральноукраїнський національний технічний університет**

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за  
другим (магістерським) рівнем вищої освіти

\_\_\_\_\_ Коваленко О.В.

*Дослідження та програмна реалізація  
системи кібербезпеки для виявлення вразливостей web-застосунків*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск

Загальна кількість аркушів: 31

Літера: РП

Кропивницький – 2021 року

```

//Модуль завантаження тестів

package project;

import org.junit.platform.runner.JUnitPlatform;
import org.junit.platform.suite.api.SelectClasses;
import org.junit.runner.RunWith;

@RunWith(JUnitPlatform.class)
@SelectClasses({
    TestSuiteCreateRent.class,
    TestSuiteCreateUser.class,
    TestSuiteCreateCar.class
})
public class RunAllTestSuits {}

// Модуль тестування безпеки проведення операції оплати за послугу аренды
// автомобіля

package project;

import core.control.DriversManager;
import core.control.Watcher;
import core.drivers.Chrome;
import core.tools.DateWorker;
import org.junit.jupiter.api.*;
import org.junit.jupiter.api.extension.ExtendWith;
import project.management.LandingPage;
import project.management.ManagementLoginPage;
import project.management.entities.RentFormTestData;
import project.management.entities.CarEntity;
import project.management.entities.CarEntity;
import project.management.car.CarsPage;
import project.management.payment.*;

import static com.codeborne.selenide.Condition.*;

@DisplayName("TEST SUITE 'Create rent'")
@ExtendWith(Watcher.class)
@TestMethodOrder(MethodOrderer.OrderAnnotation.class)
public class TestSuiteCreateRent {

    @BeforeEach
    public void beforeExecutionTests() {
        DriversManager.launch(new Chrome().setup());
    }

    @AfterAll
    public static void afterExecutionTests() {
        DriversManager.quitAll();
    }

    @Order(1)
    @DisplayName("TEST CASE: Create Rent (Stripe, Without Source)")
    @Test
    public void createRentStripeWithoutSource() {
        ManagementLoginPage.open()
            .login("testCarRent123+1211197@gmail.com",
                System.getenv("TcicarrentalRentTestPass"));
        CarsPage carsPage = CarsPage.open().renderCarList();;

        CarEntity carEntity = new CarEntity(
            "https://www.google.com",

```

```
"https://beta.t.tcicarrentalengines.us/rails/active_storage/blobs/eyJfcmFpbHMiOnsibWVz" +
```

```
"c2FnZSI6IkJBaHBBaElHIiwizXhwIjpuZDVsLCJwdXIiOiJibG9ix2lkIn19--c648464d7f193d26cb0ebd91" +
```

```
"91809cd93a1433a7/car_exterior.usg"
```

```
);
CarEntity carEntityD = new CarEntity(
    "Queen (en) [auto] (Diesel)",
    "60,000", null
);
```

```
CarEntity carEntity = new CarEntity(
    "Queen (en) [auto] (Gas)",
    "60,000", null
);
```

```
String price = carEntity.priceWithInsurance;
```

```
carsPage.getCarItem(0)
    .check(
        carEntity.name, carEntity.zipCode, carEntity.city,
carEntity.address,
        carEntity.interiorImageSrc
    ).getPreview().click();
```

```
LandingPage landingPage = new LandingPage();
landingPage.checkLandingHeader(carEntity.name, carEntity.address,
carEntity.city, carEntity.phone);
```

```
RentFormTestData rentTestData = new RentFormTestData(
    carEntity.carType + "", "1",
    new DateWorker().getTodayPlusDates(1) + "",
    new DateWorker().getTodayPlusDates(2) + "",
    "test name", "test@4fnerore.com", "4395494554", "test addr"
);
```

```
LandingPage.RentForm form = landingPage.getRentForm();
```

```
form.getErrorMessage().shouldBe(not(visible));
```

```
form.getCarType().select(rentTestData.carType);
form.getGuestsCount().select(rentTestData.guests);
form.getFrom().select(rentTestData.From);
form.getTo().select(rentTestData.To);
```

```
form.getPlanInfo().getSedanPlan().check("Insurance included", price);
form.getPlanInfo().getNext()
    .check(price)
    .click();
```

```
form.getUserInfo().fill(
    rentTestData.username, rentTestData.email,
    rentTestData.phone, rentTestData.address
);
```

```
form.getBookResult()
    .check("Insurance included", price, 1, "Taxes are included")
    .getBookNow().click();
```

```
PaymentPage paymentPage = new PaymentPage();
paymentPage.getReservationDetails().check(
    rentTestData.username, rentTestData.phone, rentTestData.address,
    rentTestData.From, rentTestData.To, rentTestData.guests
);
```

```
paymentPage.getPaymentDetails()
    .check(price, true)
    .getPurchase().click();
```



```

LandingPage.RentForm form = landingPage.getRentForm();

form.getErrorMessage().shouldBe(not(visible));

form.getCarType().select(rentTestData.carType);
form.getGuestsCount().select(rentTestData.guests);
form.getFrom().select(rentTestData.From);
form.getTo().select(rentTestData.To);

form.getPlanInfo().getPlanList().getInsuranceIncluded().checkPrice(carEntity.priceWithInsurance);
form.getPlanInfo().getPlanList().getWithoutInsurance().checkPrice(price).click();
form.getPlanInfo().getPriceAlert().shouldHave(text("All prices are tax included"));
form.getPlanInfo().getNext().click();

form.getUserInfo().fill(
    rentTestData.username, rentTestData.email,
    rentTestData.phone, rentTestData.address
);
form.getBookResult().check("Without insurance", price, 2, "Taxes are included").getBookNow().click();

PaymentPage paymentPage = new PaymentPage();
paymentPage.getReservationDetails().check(
    rentTestData.username, rentTestData.phone, rentTestData.address,
    rentTestData.From, rentTestData.To, rentTestData.guests
);
paymentPage.getPaymentDetails().check(price, true).getPurchase().click();

StripePaymentPage StripePaymentPage = StripePaymentPage.render();

StripePaymentPage.getApplicationInformation().check(rentTestData.username, rentTestData.email);
StripePaymentPage.getNameOfCar().shouldHave(text(carEntity.name));
StripePaymentPage.getPlanDetails().check(
    carEntity.carType.replace(" (No Diesel)", ""), "Without insurance"
);

StripePaymentPage.getGuestDetails().getEea().widget().selectOption("Yes");
StripePaymentPage.getGuestDetails().check(
    rentTestData.username + "",
    rentTestData.address + "",
    rentTestData.phone + "",
    rentTestData.email + "",
    "Yes"
);
StripePaymentPage.getRentDetails().check(
    rentTestData.From + "",
    rentTestData.To + "",
    2, "15:00"
);

StripePaymentPage.getTourPrice(); //todo
StripePaymentPage.getPaymentSection().fill("4111111111111111", "09", "2021", "946");
StripePaymentPage.getAttentionInformation(); //todo
}

@Order(3)
@DisplayName("TEST CASE: Create Rent (STRIPE, Without Source)")
@Test

```

```

public void createRentSTRIPEWithoutSource() {
    ManagementLoginPage.open()
        .login("testCarRent123+1211193@gmail.com",
System.getenv("TcicarrentalRentTestPass"));

    CarEntity carEntity = new CarEntity(
        "Saffron Castle Resort (en) [auto]",
        "4-chōme-2 Toranomom (en)",
        "LA", "105-0001",
        "9945959995", "https://www.google.com",

"https://beta.t.tcicarrentalengines.us/rails/active_storage/blobs/" +

"eyJfcmFpbHMiOnsibWVzc2FnZSI6IkJBaHBBdDhGIiwizXhwIjpudWxsLCJwdXIiOiJibG9iX2lkIn19-" +

        "-68aca82c9a0e9a929fe3bc3cb795f55417cd3af8/exterior2.usg"
    );
    CarEntity carEntity = new CarEntity(
        "Classic roop (en) (Diesel)",
        "2,150", null
    );
    String price = carEntity.priceWithInsurance;

    CarsPage carsPage = new CarsPage().renderCarList();
    carsPage.getCarItem(0)
        .check(
            carEntity.name, carEntity.zipCode, carEntity.city,
carEntity.address,
            carEntity.interiorImageSrc
        ).getPreview().click();

    LandingPage landingPage = new LandingPage();
    landingPage.checkLandingHeader(carEntity.name, carEntity.address,
carEntity.city, carEntity.phone);

    RentFormTestData rentTestData = new RentFormTestData(
        carEntity.carType + "", "1",
        new DateWorker().getTodayPlusDates(1) + "",
        new DateWorker().getTodayPlusDates(2) + "",
        "test name", "test@4fnerore.com", "4395494554", "test addr"
    );

    LandingPage.RentForm form = landingPage.getRentForm();

    form.getErrorMessage().shouldBe(not(visible));

    form.getCarType().select(rentTestData.carType);
    form.getGuestsCount().select(rentTestData.guests);
    form.getFrom().select(rentTestData.From);
    form.getTo().select(rentTestData.To);

    form.getPlanInfo().getSedanPlan().check("Without insurance", price);
    form.getPlanInfo().getPriceAlert().shouldHave(text("All prices are tax
included"));
    form.getPlanInfo().getNext()
        .check(price)
        .click();

    form.getUserInfo().fill(
        rentTestData.username, rentTestData.email,
        rentTestData.phone, rentTestData.address
    );
    form.getBookResult()
        .check("Without insurance", price, 1, "Taxes are included")
        .getBookNow().click();

    PaymentPage paymentPage = new PaymentPage();
    paymentPage.getReservationDetails().check(
        rentTestData.username, rentTestData.phone, rentTestData.address,

```

```

        rentTestData.From, rentTestData.To, rentTestData.guests
    );
    paymentPage.getPaymentDetails()
        .check(price, true)
        .getPurchase().click();
    StripePaymentPopup Stripe = StripePaymentPopup.render();
    Stripe
        .fill("test", "1234 3245 1111 2435",
            "March", "2022", "123")
        .check("test", "1234 3245 1111 2435",
            "March", "2022", "123", price)
        .getPayWithCard().click();

    SuccessPaymentPage successPaymentPage = SuccessPaymentPage.render();
    successPaymentPage.check(
        rentTestData.username + "",
        rentTestData.From + "",
        rentTestData.To + "",
        rentTestData.carType.replace(" (Disel)", "") + "",
        "Without insurance",
        rentTestData.guests + "",
        price + "",
        rentTestData.email + ""
    );
}

@Order(4)
@DisplayName("TEST CASE: Create Rent (STRIPE, TCI)")
@Test
public void createRentStripeTCI() {
    ManagementLoginPage.open()
        .login("testlogin83+0400@gmail.com ",
System.getenv("TcicarrentalRentOlgaPass"));
    //todo test
}

@Order(5)
@DisplayName("TEST CASE: Create Rent (Stripe, TCI)")
@Test
public void createRentStripeTCI() {
    ManagementLoginPage.open()
        .login("testCarRent123+1510191@gmail.com",
System.getenv("TcicarrentalRentTestPass"));
    //todo test
}

@Order(6)
@DisplayName("TEST CASE: Create Rent (STRIPE, TCI)")
@Test
public void createRentStripeTCI() {
    ManagementLoginPage.open()
        .login("testlogin83+04@gmail.com ",
System.getenv("TcicarrentalRentOlgaPass"));
    //todo test
}
}

```

// Модуль перевірки безпеки додавання картки автомобіля для аренди

```
package project;
```

```

import com.codeborne.selenide.Selenide;
import core.control.DriversManager;
import core.control.Watcher;
import core.control.application_tools.LocalStorage;

```

```

import core.drivers.Chrome;
import org.junit.jupiter.api.*;
import org.junit.jupiter.api.extension.ExtendWith;
import project.management.ManagementLoginPage;
import project.management.ManagementSidebar;
import project.management.car.ConnectToChannelManagerPopup;
import project.management.car.CarNewPage;

import java.io.File;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Random;

@DisplayName("TEST SUITE 'Create car'")
@ExtendWith(Watcher.class)
public class TestSuiteCreateCar {

    @BeforeAll
    public static void beforeExecutionTests() {
        DriversManager.launch(new Chrome().setup());
    }

    @AfterEach
    public void afterEachExecutionTest() {
        LocalStorage.getStorage().setItem("management", "");
    }

    @AfterAll
    public static void afterExecutionTests() {
        DriversManager.quitAll();
    }

    @DisplayName("TEST CASE: Create new car")
    @Test
    public void createNewCar() {
        ManagementLoginPage.open()
            .login("testlogin83+0400@gmail.com",
                System.getenv("TcicarrentalRentOlgaPass"));
        new ManagementSidebar().getAddCar().click();

        ConnectToChannelManagerPopup
            .render()
            .fill("B3892756",
                System.getenv("TCIApiKeyPass")).getConfirm().click();
        Selenide.sleep(3000);

        String code = " #" + String.format("%06d", new
            Random().nextInt(999999));

        CarNewPage carNewPage = new CarNewPage();
        CarNewPage.RegisterCar registerCar = carNewPage.getRegisterCar();
        registerCar.fill(
            "Parallel Palms Car (en) [auto]" + code,
            "Parallel Palms Car (en) [auto]" + code,
            "P48909",
            new File("src/main/resources/" + "images/logos/car_logo.useg"),
            new File("src/main/resources/" +
                "images/exterior/exterior_car.usg"),
            "0100973", "Ford",
            "4-chôme-6 Ford (en)",
            "4-chôme-6 Ford (en)",
            "5495959444",
            "Privacy policy (en)", "Privacy policy (en)"
        );
        registerCar.getSaveAndAddCancellationPolicy().click();
    }
}

```

```

        CarNewPage.CancellationSettings cancellationSettings =
carNewPage.getCancellationSettings();
        cancellationSettings.fillEarlyCancellationFeeSettings(
            "3", "5", "10"
        );
        cancellationSettings.getSaveAndAddCancellationPolicy().click();

        CarNewPage.AddCar addCar = carNewPage.getAddCar();
        addCar.fill(
            "Double (en)" + code,
            "Double (en)" + code,
            false,
            "Double",
            "4",
            "2",
            true,
            "17074713",
            false,
            null,
            new ArrayList<>(Arrays.asList("TV", "Comb")), // these amenities
will be exclude
            "Car description (en)",
            "Car description (en)",
            new File("src/main/resources/" +
"images/vip_car/vip_car_1.usg"),
            new File("src/main/resources/" + "images/vip_car/vip_car_2.usg")
        );
        addCar.getComplete().click();

        //todo check toast message
        //todo check that car is successfully created
    }
}

```

//Модуль перевірки створення картки нового користувача сайту

```
package project;
```

```

import core.control.DriversManager;
import core.control.Watcher;
import core.drivers.Chrome;
import core.tools.MailController;
import org.junit.jupiter.api.*;
import org.junit.jupiter.api.extension.ExtendWith;
import project.admin.AddNewClientPopup;
import project.admin.AdminLoginPage;
import project.admin.AdminSidebar;
import project.admin.emails.SingUpNewClientEmail;
import project.management.CompleteSingUpPage;
import project.management.ConnectStripePage;
import project.management.car.CarsPage;

```

```

import java.util.Objects;
import java.util.Random;

```

```

@DisplayName("TEST SUITE 'Create new User'")
@ExtendWith(Watcher.class)
@TestMethodOrder(MethodOrderer.OrderAnnotation.class)
public class TestSuiteCreateUser {
    private String code;
    private String email;

    @BeforeEach
    public void beforeEachExecutionTest() {
        DriversManager.launch(new Chrome().setup());
        this.code = " #" + String.format("%06d", new Random().nextInt(999999));
    }
}

```

```

        this.email = MailController.EMAIL
            .replace("@", code.replace(" #", "+") + "@");
    }

    @AfterEach
    public void afterEachExecutionTest() {
        MailController.getUnreadMails(); // marks all mails as read
    }

    @AfterAll
    public static void afterExecutionTests() {
        DriversManager.quitAll();
    }

    @DisplayName("TEST CASE: Create new client (Stripe, TL TCI)")
    @Order(1)
    @Test
    public void createNewClientStripeTCI() {
        AdminLoginPage.open()
            .login("default-admin@mail.com", "default");

        new AdminSidebar().getAddNewClient().click();
        AddNewClientPopup popup = AddNewClientPopup.render();
        popup.fill(
            "Devbug" + code,
            "test_" + code.replace(" ", ""),
            email + "",
            "4395995495",
            "test google (en)" + code,
            "test google (jp)" + code,
            "TL TCI",
            popup.fillStripeData(true),
            true
        ).getAddNewClient().click();

        SingUpNewClientEmail mail = new SingUpNewClientEmail(Objects
            .requireNonNull(MailController.waitNewMail(120)));
        mail.check(email, "test_" + code.replace(" ", ""));

        CompleteSingUpPage completeSingUpPage =
        CompleteSingUpPage.open(mail.getInviteLink());
        String password = System.getenv("TcicarrentalRentTestPass");
        completeSingUpPage
            .fill(password + "_Te$1", password + "_Te$1")
            .getSetPassword().click();
        ConnectStripePage.render();
        new PageHeader().getLogout().click();
    }

    @DisplayName("TEST CASE: Create new client (STRIPE, Without Source)")
    @Order(2)
    @Test
    public void createNewClientStripeWithoutSource() {
        AdminLoginPage.open()
            .login("default-admin@mail.com", "default");

        new AdminSidebar().getAddNewClient().click();
        AddNewClientPopup popup = AddNewClientPopup.render();
        popup.fill(
            "Devbug2" + code,
            "test2_" + code.replace(" ", ""),
            email + "",
            "95495423949",
            "test google 2 (en)" + code,
            "test google 2 (jp)" + code,
            "Without source",
            popup.fillStripeData("10"),

```

```

        true
    ).getAddNewClient().click();

    SingUpNewClientEmail mail = new SingUpNewClientEmail(Objects
        .requireNonNull(MailController.waitNewMail(120)));
    mail.check(email, "test2_" + code.replace(" ", ""));

    CompleteSingUpPage completeSingUpPage =
CompleteSingUpPage.open(mail.getInviteLink());
    String password = System.getenv("TcicarrentalRentTestPass");
    completeSingUpPage
        .fill(password + "_Te$1", password + "_Te$1")
        .getSetPassword().click();
    new CarsPage();
    new PageHeader().getLogout().click();
}

@Test
@Order(3)
@DisplayName("TEST CASE: Create new client (STRIPE, TCI)")
@Test
public void createNewClientStripeTCI() {
    AdminLoginPage.open()
        .login("default-admin@mail.com", "default");

    new AdminSidebar().getAddNewClient().click();
    AddNewClientPopup popup = AddNewClientPopup.render();
    popup.fill(
        "Devbug3" + code,
        "test3_" + code.replace(" ", ""),
        email + "",
        "59955404230",
        "test google 3 (en)" + code,
        "test google 3 (jp)" + code,
        "TL TCI",
        popup.fillStripeData(
            "6",
            popup.fillLegalInformationJpData(
                "test seller (en)",
                "test representative (en)",
                "test address (en)",
                "59955404230",
                email + "",
                "21000",
                "test surcharges (en)",
                "test payment method (en)",
                "3",
                "7",
                "5"
            ),
            popup.fillLegalInformationEnData(
                "test seller (en)",
                "test representative (en)",
                "test address (en)",
                "21000",
                "test surcharges (en)",
                "test payment method (en)",
                "3",
                "7",
                "5"
            )
        ),
        true
    ).getAddNewClient().click();

    SingUpNewClientEmail mail = new SingUpNewClientEmail(Objects
        .requireNonNull(MailController.waitNewMail(120)));
    mail.check(email, "test2_" + code.replace(" ", ""));
}

```

```

        CompleteSingUpPage completeSingUpPage =
CompleteSingUpPage.open(mail.getInviteLink());
String password = System.getenv("TcicarrentalRentTestPass");
completeSingUpPage
    .fill(password + "_Te$1", password + "_Te$1")
    .getSetPassword().click();
new CarsPage();
new PageHeader().getLogout().click();
    }
}

// Модуль тестування безпеки операції логіну користувача

package project;

import core.control.Delay;
import core.main_web_elements.Button;
import core.main_web_elements.Input;
import core.main_web_elements.WebElement;
import io.qameta.allure.Step;
import org.openqa.selenium.By;

import static com.codeborne.selenide.Condition.visible;

public class LoginForm extends WebElement {
    private Input email;
    private Input password;
    private Button submit;
    private Button forgotPassword;
    private WebElement rememberMe;

    public LoginForm() {
        super("//div[@class='md-title' and text()='Login']/..");
        this.email = new
Input(this.widget().$(By.xpath("form//input[@type='email']")));
        this.password = new
Input(this.widget().$(By.xpath("form//input[@type='password']"))) {
            @Step("☒ set value [*****]")
            @Override
            public WebElement setValue(String text) {
                return super.setValue(text);
            }
        };
        this.submit = new
Button(this.widget().$(By.xpath("form//button[@type='submit']")));
        this.forgotPassword = new
Button(this.widget().$(By.xpath("div//a[text()='Forgot password?']")));
        this.rememberMe = new WebElement(this.widget().$(By.xpath(""))); //todo
checkbox
    }

    public void render() {
        this.widget().waitUntil(visible, Delay.STANDARD.getTime());
        this.email.shouldBe(visible);
        this.password.shouldBe(visible);
        this.submit.shouldBe(visible);
        this.forgotPassword.shouldBe(visible);
    }

    @Step("email")
    public Input getEmail() {
        return email;
    }

    @Step("password")

```

```

public Input getPassword() {
    return password;
}

@Step("login (button)")
public Button getSubmit() {
    return submit;
}

@Step("forgot password")
public Button getForgotPassword() {
    return forgotPassword;
}

@Step("remember me")
public WebElement getRememberMe() {
    return rememberMe;
}
}

package project.management;

import com.codeborne.selenide.Selenide;
import core.control.Delay;
import core.control.Domen;
import core.main_web_elements.WebElement;
import io.qameta.allure.Step;
import project.admin.AdminLoginPage;
import project.management.car.CarsPage;

import static com.codeborne.selenide.Condition.visible;

public class ManagementLoginPage extends AdminLoginPage {
    private WebElement logo;

    @Step("open [Management Login page]")
    public static ManagementLoginPage open() {
        return Selenide.open(ManagementLoginPage.url(),
ManagementLoginPage.class);
    }

    public static String url() {
        return Domen.MANAGEMENT.getUrl().concat("login");
    }

    @Override
    public void render() {
        super.render();
        this.logo = new WebElement("//img[@alt='Tcicarrental Rent']");
        logo.widget().shouldBe(visible);
    }

    @Step("logo")
    public WebElement getLogo() {
        return logo;
    }

    @Override
    public void login(String email, String password) {
        getLoginForm().getEmail().setValue(email);
        getLoginForm().getPassword().setValue(password);
        getLoginForm().getSubmit().click();
        new CarsPage().getTitle().widget().waitUntil(visible,
Delay.STANDARD.getTime());
        Selenide.sleep(Delay.LOW_1000.getTime());
    }
}

```

```
    }  
}  
  
//Модуль тестування landing бронювання обраної марки авто  
  
package project.management;  
  
import com.codeborne.selenide.Selenide;  
import com.codeborne.selenide.SelenideElement;  
import core.control.Delay;  
import core.main_web_elements.*;  
import io.qameta.allure.Step;  
import org.openqa.selenium.By;  
  
import static com.codeborne.selenide.Condition.*;  
import static com.codeborne.selenide.Selenide.$;  
  
public class LandingPage {  
  
    private WebElement rentTitle = new WebElement("//div[@class='rent-  
title']/h2");  
    private WebElement address = new WebElement("//div[@class='rent-  
car']/div[1]");  
    private WebElement phone = new WebElement("//div[@class='rent-  
car']/div[2]");  
    private WebElement website = new WebElement("//div[@class='rent-  
car']/div[3]");  
    private RentForm rentForm;  
  
    public LandingPage() {  
        render();  
    }  
  
    public void render() {  
        rentTitle.waitUntil(visible, Delay.LONG.getTime());  
        this.rentForm = new RentForm();  
    }  
  
    @Step("Rent title")  
    public WebElement getRentTitle() {  
        return rentTitle;  
    }  
  
    @Step("Address")  
    public WebElement getAddress() {  
        return address;  
    }  
  
    @Step("Phone")  
    public WebElement getPhone() {  
        return phone;  
    }  
  
    @Step("Website")  
    public WebElement getWebsite() {  
        return website;  
    }  
  
    public RentForm getRentForm() {  
        return rentForm;  
    }  
}
```

```

    public void checkLandingHeader(String title, String address, String city,
String phone) {
        getRentTitle().shouldHave(text(title));
        getAddress().shouldHave(text(address + ", " + city));
        getPhone().shouldHave(text(phone));
    }

```

```

public class RentForm extends WebElement {
    private Selector carType;
    private Selector guestsCount;
    private DatePicker From;
    private DatePicker To;
    private UserInfo userInfo;
    private PlanInfo planInfo;
    private BookResult bookResult;
    private WebElement errorMessage;

    public RentForm() {
        super("//div[@class='rent-form']/form");
        this.carType = new Selector("//label[text()='Car type']/following-
sibling::div");
        this.guestsCount = new Selector("//label[text()='Guests']/following-
sibling::div");
        this.From = new DatePicker("//label[text()='Check In']/preceding-
sibling::input");
        this.To = new DatePicker("//label[text()='Check Out']/preceding-
sibling::input") {

            @Override
            public void select(String date) {
                super.select(date);
                Selenide.sleep(Delay.LOW_500.getTime());
                if (errorMessage.widget().is(not(visible)) &&
                    errorMessage.widget().is(not(text("Requested period
is not available to book")))) {
                    RentForm.this.planInfo = new PlanInfo();
                }
            }
        };
        this.errorMessage = new WebElement("//p[@class='md-error']");
        render();
    }

    public void render() {
        this.widget().waitUntil(visible, Delay.STANDARD.getTime());
        if (errorMessage.widget().is(not(visible)) &&
            errorMessage.widget().is(not(text("Sorry, there are no cars
matching your criteria. " +
                "Try to change date range or number of guests")))) {
            this.planInfo = new PlanInfo();
        }
        this.userInfo = new UserInfo();
        this.bookResult = new BookResult();
    }

    @Step("Car type")
    public Selector getCarType() {
        return carType;
    }

    @Step("Guests count")
    public Selector getGuestsCount() {
        return guestsCount;
    }
}

```

```

@Step("Check In")
public DatePicker getFrom() {
    return From;
}

@Step("Check Out")
public DatePicker getTo() {
    return To;
}

public UserInfo getUserInfo() {
    return userInfo;
}

@Step("Error message")
public WebElement getErrorMessage() {
    return errorMessage;
}

public PlanInfo getPlanInfo() {
    return planInfo;
}

public BookResult getBookResult() {
    return bookResult;
}

public class PlanInfo extends WebElement {
    private SedanPlan sedanPlan;
    private PlanList planList;

    private ButtonNext next;
    private WebElement priceAlert;

    public PlanInfo() {
        super("//div[contains(@class, 'confirm-button-container')]");
        this.priceAlert = new
WebElement(this.widget().$(By.xpath("div[@class='price-alert']")));
        this.next = new ButtonNext(this.widget().$(By.xpath("button")))
    {
        @Override
        public void click() {
            super.click();
            RentForm.this.planInfo.widget().waitUntil(hidden,
Delay.LOW_2000.getTime());
            RentForm.this.userInfo = new UserInfo();
            RentForm.this.bookResult = new BookResult();
        }
    };
        renderPlanSection();
    }

    private void renderPlanSection() {
        Selenide.sleep(Delay.LOW_500.getTime());
        $(By.xpath("//div[@class='spinner']")).waitUntil(hidden,
Delay.STANDARD.getTime());
        this.singlePlan = new SinglePlan();
        this.planList = new PlanList();
    }

    @Step("Single Plan")
    public SinglePlan getSinglePlan() {

```

```

        return singlePlan;
    }

    @Step("Both Plan list")
    public PlanList getPlanList() {
        return planList;
    }

    @Step("Next")
    public ButtonNext getNext() {
        return next;
    }

    @Step("Price alert (tax)")
    public WebElement getPriceAlert() {
        return priceAlert;
    }

    public class ButtonNext extends Button {

        private ButtonNext(SelenideElement element) {
            super(element);
        }

        @Step("price")
        public ButtonNext check(String price) {
            getNext().shouldHave(text("Next " + price));

            return this;
        }
    }

    public class SinglePlan extends WebElement {
        private WebElement planType;
        private WebElement price;

        private SinglePlan() {
            super("//div[contains(@class, 'plan-list')]");
            this.planType = new
WebElement(this.widget().$(By.xpath("div/div[@class='single-plan']")));
            this.price = new
WebElement(this.widget().$(By.xpath("div/div[@class='single-price']")));
        }

        @Step("plan type")
        public WebElement getPlanType() {
            return planType;
        }

        @Step("price")
        public WebElement getPrice() {
            return price;
        }

        public SinglePlan check(String planType, String price) {
            getPlanType().shouldHave(text(planType));
            getPrice().shouldHave(text(price));
            return this;
        }
    }
}

```

```

public class PlanList extends WebElement {
    private BookItemButton withoutInsurance;
    private BookItemButton insuranceIncluded;

    private PlanList() {
        super("//div[contains(@class, 'plan-
list')]/div[@class='both-plan']");
        this.withoutInsurance = new BookItemButton(this.widget()
            .$(By.xpath("div[contains(text(), 'Without
insurance')]")));
        this.insuranceIncluded = new BookItemButton(this.widget()
            .$(By.xpath("div[contains(text(), 'Insurance
included')]")));
    }

    @Step("Without Insurance (button)")
    public BookItemButton getWithoutInsurance() {
        return withoutInsurance;
    }

    @Step("Insurance Included (button)")
    public BookItemButton getInsuranceIncluded() {
        return insuranceIncluded;
    }

    public class BookItemButton extends WebElement {
        private WebElement price;

        private BookItemButton(SelenideElement element) {
            super(element);
            this.price = new
WebElement(this.widget().$(By.xpath("span")));
        }

        @Step("price")
        public WebElement getPrice() {
            return price;
        }

        public BookItemButton checkPrice(String price) {
            getPrice().shouldHave(text(price));
            return this;
        }
    }
}

public class UserInfo extends WebElement {
    private Input name;
    private Input email;
    private Input phoneNumber;
    private Input address;

    private UserInfo() {
        super("//div[@class='user-info']");
        this.name = new
Input(this.widget().$(By.xpath("div//label[text()='Name']/following-
sibling::input")));
    }
}

```

```

        this.email = new
Input(this.widget().$(By.xpath("div//label[text()='Email']/following-
sibling::input")));
        this.phoneNumber = new
Input(this.widget().$(By.xpath("div//label[text()='Phone number']/following-
sibling::input")));
        this.address = new
Input(this.widget().$(By.xpath("div//label[text()='Address']/following-
sibling::input")));
    }

```

```

    @Step("Name")
    public Input getName() {
        return name;
    }
    @Step("Email")
    public Input getEmail() {
        return email;
    }

    @Step("Phone number")
    public Input getPhoneNumber() {
        return phoneNumber;
    }

    @Step("Address")
    public Input getAddress() {
        return address;
    }

```

```

    public void fill(String name, String email, String phoneNumber,
String address) {
        getName().setValue(name);
        getEmail().setValue(email);
        getPhoneNumber().setValue(phoneNumber);
        getAddress().setValue(address);
    }
}

```

```

public class BookResult extends WebElement {
    private WebElement nights;
    private WebElement price;
    private WebElement planType;
    private WebElement taxIndicator;
    private Button cancel;
    private Button bookNow;

    private BookResult() {
        super($(By.xpath("//div[@class='rent-form-footer between rent-
info-final']")));
        this.nights = new
WebElement(this.widget().$(By.xpath("div//div[@class='rent-nights']")));
        this.price = new
WebElement(this.widget().$(By.xpath("div//div[@class='price']")));
        this.planType = new
WebElement(this.widget().$(By.xpath("div//div[@class='plan-type']")));
        this.taxIndicator = new
WebElement(this.widget().$(By.xpath("div//div[@class='tax-indicator']")));
        this.cancel = new Button(this.widget().$(By.xpath("following-
sibling::div//div[text()='Back']/../.."))); {

        @Override
        public void click() {
            super.click();
        }
    }
}

```



```
public class AdminLoginPage {
    private LoginForm loginForm;

    public AdminLoginPage() {
        render();
    }

    @Step("open [Admin Login page]")
    public static AdminLoginPage open() {
        return Selenide.open(AdminLoginPage.url(), AdminLoginPage.class);
    }

    public static String url() {
        return Domen.ADMIN.getUrl().concat("login");
    }

    public void render() {
        if (getStorage().getItem("locale") == null) {
            getStorage().setItem("locale", "en");
            Selenide.refresh();
        }
        this.loginForm = new LoginForm();
        loginForm.render();
    }

    public LoginForm getLoginForm() {
        return loginForm;
    }

    public void login(String email, String password) {
        getLoginForm().getEmail().setValue(email);
        getLoginForm().getPassword().setValue(password);
        getLoginForm().getSubmit().click();
        new ClientsPage();
    }
}
```

```

//Модуль тестування операції додавання нового користувача в систему
//з правами адміністратора
package project.admin;

import core.control.Delay;
import core.main_web_elements.*;
import io.qameta.allure.Step;

import static com.codeborne.selenide.Condition.visible;
import static com.codeborne.selenide.Selenide.$x;

public class AddNewClientPopup extends WebElement {
    private WebElement title;
    private Button btnClose;
    private Input companyName;
    private Input username;
    private Input email;
    private Input phoneNumber;
    private Input googleDisplayNameEn;
    private Input googleDisplayNameJp;
    private Selector source;
    private Selector paymentSystem;
    private Checkbox sendInvitationEmail;
    private Button addNewClient;

    private Checkbox USCarAssociationMemberCars;
    private Input tcicarrentalRentCommission;
    private Button next;
    private LegalInformationJa legalInformationJa;
    private LegalInformationEn legalInformationEn;

    public static AddNewClientPopup render() {
        $x("//span[text()='Add New Client']/../..").waitUntil(visible,
            Delay.MEDIUM.getTime());
        return new AddNewClientPopup();
    }

    private AddNewClientPopup() {
        super("//span[text()='Add New Client']/../..");
        this.title = new WebElement("//span[text()='Add New Client']");
        this.btnClose = new Button(this.widget().$x("span/a"));
        this.companyName = new
            Input(this.widget().$x("div//label[text()='Company Name']/following-
                sibling::input"));
        this.username = new
            Input(this.widget().$x("div//label[text()='Username']/following-
                sibling::input"));
        this.email = new
            Input(this.widget().$x("div//label[text()='Email']/following-sibling::input"));
        this.phoneNumber = new Input(this.widget().$x("div//label[text()='Phone
            number']/following-sibling::input"));
        this.googleDisplayNameEn = new Input(this.widget()
            .$x("div//label[text()='Google display name (en)']/following-
                sibling::input"));
        this.googleDisplayNameJp = new Input(this.widget()
            .$x("div//label[text()='Google display name (jp)']/following-
                sibling::input"));
        this.source = new
            Selector(this.widget().$x("div//label[text()='Source']/following-
                sibling::div"));
        this.paymentSystem = new
            Selector(this.widget().$x("div//label[text()='Payment system']/following-
                sibling::div")) {

                @Override
                public void select(String option) {

```

```

        super.select(option);
        AddNewClientPopup.this.renderPaymentSystem();
    }
};
renderPaymentSystem();
this.sendInvitationEmail = new
Checkbox(this.widget().$x("//div/label[text()='Send invitation email']/.."));
this.addNewClient = new
Button(this.widget().$x("div//button//div[text()='Add New Client']/../.."));
}

private void renderPaymentSystem() {
    switch (paymentSystem.getInput().widget().getValue()) {
        case StripeData.PAYMENT_SYSTEM :
            AddNewClientPopup.this.USCarAssociationMemberCars = new
Checkbox(
                "//label[text()='US Car Association member cars (stripe
fee 2.5%')/.."];
            break;
        case StripeData.PAYMENT_SYSTEM :
            AddNewClientPopup.this.tcicarrentalRentCommission = new
Input(AddNewClientPopup.this.widget()
                .$x("div//label[text()='Tcicarrental Rent
Commission']/following-sibling::input"));
            break;
        case StripeData.PAYMENT_SYSTEM :
            AddNewClientPopup.this.tcicarrentalRentCommission = new
Input(AddNewClientPopup.this.widget()
                .$x("div//label[text()='Tcicarrental Rent
Commission']/following-sibling::input"));
            AddNewClientPopup.this.next = new
Button(AddNewClientPopup.this.widget()
                .$x("div//button//div[text()='Next']/../.."));
            break;
        default:
            break;
    }
}

public void renderLegalInformationJa() {
    this.legalInformationJa = new LegalInformationJa();
}

public void renderLegalInformationEn() {
    this.legalInformationEn = new LegalInformationEn();
}

@Step("Title")
public WebElement getTitle() {
    return title;
}

@Step("X (close)")
public Button getBtnClose() {
    return btnClose;
}

@Step("Company name")
public Input getCompanyName() {
    return companyName;
}

@Step("Username")
public Input getUsername() {
    return username;
}
}

```

```

@Step("Email")
public Input getEmail() {
    return email;
}

@Step("Phone Number")
public Input getPhoneNumber() {
    return phoneNumber;
}

@Step("Google Display Name (en)")
public Input getGoogleDisplayNameEn() {
    return googleDisplayNameEn;
}

@Step("GoogleDisplayName (jp)")
public Input getGoogleDisplayNameJp() {
    return googleDisplayNameJp;
}

@Step("Source")
public Selector getSource() {
    return source;
}

@Step("Payment system")
public Selector getPaymentSystem() {
    return paymentSystem;
}

@Step("Send Invitation Email")
public Checkbox getSendInvitationEmail() {
    return sendInvitationEmail;
}

@Step("Add New Client")
public Button getAddNewClient() {
    return addNewClient;
}

@Step("US Car Association Member Cars")
public Checkbox getUSCarAssociationMemberCars() {
    return USCarAssociationMemberCars;
}

@Step("Tcicarrental Rent Commission")
public Input getTcicarrentalRentCommission() {
    return tcicarrentalRentCommission;
}

@Step("Next")
public Button getNext() {
    return next;
}

@Step("Legal Information (en)")
public LegalInformationJa getLegalInformationJa() {
    return legalInformationJa;
}

@Step("Legal Information (en)")
public LegalInformationEn getLegalInformationEn() {
    return legalInformationEn;
}

/**
 * AddNewClientPopup fill()
 * @param companyName

```

```

* @param username
* @param email
* @param phoneNumber
* @param googleDisplayNameEn
* @param googleDisplayNameJp
* @param source
* @param payment: fillStripeData(), fillStripeData(), fillStripeData()
* @return AddNewClientPopup.this
*/
public AddNewClientPopup fill(String companyName, String username, String
email, String phoneNumber,
                                String googleDisplayNameEn, String
googleDisplayNameJp, String source,
                                Payment payment, boolean
isSendInvitationEmail) {

    getCompanyName().setValue(companyName);
    getUsername().setValue(username);
    getEmail().setValue(email);
    getPhoneNumber().setValue(phoneNumber);
    getGoogleDisplayNameEn().setValue(googleDisplayNameEn);
    getGoogleDisplayNameJp().setValue(googleDisplayNameJp);
    getSource().select(source);

    if (payment instanceof StripeData) {
        getPaymentSystem().select(StripeData.PAYMENT_SYSTEM);
        getUSCarAssociationMemberCars().setFlag(((StripeData)
payment).isUSCarAssociationMemberCars);
        getSendInvitationEmail().setFlag(isSendInvitationEmail);
    } else if (payment instanceof StripeData) {
        getPaymentSystem().select(StripeData.PAYMENT_SYSTEM);
        getTcicarrentalRentCommission().setValue(((StripeData)
payment).tcicarrentalRentCommission);
        getSendInvitationEmail().setFlag(isSendInvitationEmail);
    } else if (payment instanceof StripeData) {
        getPaymentSystem().select(StripeData.PAYMENT_SYSTEM);
        getTcicarrentalRentCommission().setValue(((StripeData)
payment).tcicarrentalRentCommission);
        if (((StripeData) payment).legalInformationJpData != null) {
            getNext().click();
            renderLegalInformationJa();
            getLegalInformationJa().fill(
                ((StripeData) payment).legalInformationJpData.seller,
                ((StripeData)
payment).legalInformationJpData.representative,
                ((StripeData) payment).legalInformationJpData.address,
                ((StripeData)
payment).legalInformationJpData.sellingPrice,
                ((StripeData)
payment).legalInformationJpData.additionalSurcharges,
                ((StripeData)
payment).legalInformationJpData.paymentPeriodAndMethod,
                ((StripeData)
payment).legalInformationJpData.deliveryTime,
                ((StripeData) payment).legalInformationJpData.returns,
                ((StripeData)
payment).legalInformationJpData.cancellation,
                ((StripeData)
payment).legalInformationJpData.phoneNumber,
                ((StripeData) payment).legalInformationJpData.email
            );
            if (((StripeData) payment).legalInformationEnData != null) {
                getNext().click();
                renderLegalInformationEn();
                getLegalInformationEn().fill(
                    ((StripeData)
payment).legalInformationEnData.seller,
                    ((StripeData)
payment).legalInformationEnData.representative,

```

```

        ((StripeData)
payment).legalInformationEnData.address,
        ((StripeData)
payment).legalInformationEnData.sellingPrice,
        ((StripeData)
payment).legalInformationEnData.additionalSurcharges,
        ((StripeData)
payment).legalInformationEnData.paymentPeriodAndMethod,
        ((StripeData)
payment).legalInformationEnData.deliveryTime,
        ((StripeData)
payment).legalInformationEnData.returns,
        ((StripeData)
payment).legalInformationEnData.cancellation
    );
    getSendInvitationEmail().setFlag(isSendInvitationEmail);
}

}

}
return this;
}

public StripeData fillStripeData(boolean isUSCarAssociationMemberCars) {
    return new StripeData(isUSCarAssociationMemberCars);
}

public StripeData fillStripeData(String tcicarrentalRentCommission) {
    return new StripeData(tcicarrentalRentCommission);
}

public StripeData fillStripeData(String tcicarrentalRentCommission,
LegalInformationJpData legalInformationJpData,
    LegalInformationEnData legalInformationEnData) {
    return new StripeData(tcicarrentalRentCommission,
legalInformationJpData, legalInformationEnData);
}

public LegalInformationJpData fillLegalInformationJpData(String seller,
String representative, String address,
    String phoneNumber,
String email, String sellingPrice,
    String
additionalSurcharges, String paymentPeriodAndMethod,
    String
deliveryTime, String returns, String cancellation) {
    return new LegalInformationJpData(seller, representative, address,
phoneNumber, email, sellingPrice,
    additionalSurcharges, paymentPeriodAndMethod, deliveryTime,
returns, cancellation);
}

public LegalInformationEnData fillLegalInformationEnData(String seller,
String representative, String address,
    String
sellingPrice, String additionalSurcharges,
    String
paymentPeriodAndMethod, String deliveryTime,
    String returns,
String cancellation) {
    return new LegalInformationEnData(seller, representative, address,
sellingPrice,
    additionalSurcharges, paymentPeriodAndMethod, deliveryTime,
returns, cancellation);
}

```

```

public class LegalInformationEn extends LegalInformation {

    private LegalInformationEn() {
        super("//span[text()='Legal Information (En)']/../..");
        AddNewClientPopup.this.sendInvitationEmail = new
Checkbox(this.widget()
        .$x("//div/label[text()='Send invitation email']/.."));
        AddNewClientPopup.this.addNewClient = new Button(this.widget()
        .$x("div//button//div[text()='Add New Client']/../.."));
    }

    @Override
    public Button getBack() {
        return new Button(super.getBack().widget()) {

            @Override
            public void click() {
                super.click();
                AddNewClientPopup.this.renderLegalInformationJa();
            }
        };
    }
}

public class LegalInformationJa extends LegalInformation {
    private Input phoneNumber;
    private Input email;

    private LegalInformationJa() {
        super("//span[text()='Legal Information (Jp)']/../..");
        this.phoneNumber = new
Input(this.widget() .$x("div//label[text()='Phone number']/following-
sibling::input"));
        this.email = new
Input(this.widget() .$x("div//label[text()='Email']/following-sibling::input"));
        AddNewClientPopup.this.next = new
Button(this.widget() .$x("div//button//div[text()='Next']/../.."));
    }

    @Step("Phone Number")
    public Input getPhoneNumber() {
        return phoneNumber;
    }

    @Step("Email")
    public Input getEmail() {
        return email;
    }

    @Override
    public Button getBack() {
        return new Button(super.getBack().widget()) {

            @Override
            public void click() {
                super.click();
                AddNewClientPopup.render();
            }
        };
    }
}

```

```

        public void fill(String seller, String representative, String address,
String sellingPrice,
                        String additionalSurcharges, String
paymentPeriodAndMethod, String deliveryTime,
                        String returns, String cancellation, String
phoneNumber, String email) {

            super.fill(seller, representative, address, sellingPrice,
additionalSurcharges, paymentPeriodAndMethod,
                    deliveryTime, returns, cancellation);
            getPhoneNumber().setValue(phoneNumber);
            getEmail().setValue(email);
        }
    }
}

```

```

public static class LegalInformation extends WebElement {
    private Input seller;
    private Input representative;
    private Input address;
    private Input sellingPrice;
    private Input additionalSurcharges;
    private Input paymentPeriodAndMethod;
    private Input deliveryTime;
    private Input returns;
    private Input cancellation;
    private Button back;

    public LegalInformation(String xpath) {
        super(xpath);
        this.seller = new
Input(this.widget().$x("div//label[text()='Seller']/following-sibling::input"));
        this.representative = new
Input(this.widget().$x("div//label[text()='Representative']/following-
sibling::input"));
        this.address = new
Input(this.widget().$x("div//label[text()='Address']/following-
sibling::input"));
        this.sellingPrice = new
Input(this.widget().$x("div//label[text()='Selling price']/following-
sibling::input"));
        this.additionalSurcharges = new
Input(this.widget().$x("div//label[text()='Additional Surcharges']/following-
sibling::input"));
        this.paymentPeriodAndMethod = new
Input(this.widget().$x("div//label[text()='Payment Period and
Method']/following-sibling::textarea"));
        this.deliveryTime = new
Input(this.widget().$x("div//label[text()='Delivery Time']/following-
sibling::input"));
        this.returns = new
Input(this.widget().$x("div//label[text()='Returns']/following-
sibling::input"));
        this.cancellation = new
Input(this.widget().$x("div//label[text()='Cancellation']/following-
sibling::textarea"));
        this.back = new
Button(this.widget().$x("div//button/div[text()='Back']/../.."));
    }

    @Step("Seller")
    public Input getSeller() {
        return seller;
    }
}

```

```

@Step("Representative")
public Input getRepresentative() {
    return representative;
}

@Step("Address")
public Input getAddress() {
    return address;
}

@Step("Selling Price")
public Input getSellingPrice() {
    return sellingPrice;
}

@Step("Additional Surcharges")
public Input getAdditionalSurcharges() {
    return additionalSurcharges;
}

@Step("Payment Period And Method")
public Input getPaymentPeriodAndMethod() {
    return paymentPeriodAndMethod;
}

@Step("Delivery Time")
public Input getDeliveryTime() {
    return deliveryTime;
}

@Step("Returns")
public Input getReturns() {
    return returns;
}

@Step("Cancellation")
public Input getCancellation() {
    return cancellation;
}

@Step("Back")
public Button getBack() {
    return back;
}

public void fill(String seller, String representative, String address,
String sellingPrice,
                String additionalSurcharges, String
paymentPeriodAndMethod, String deliveryTime,
                String returns, String cancellation) {

    getSeller().setValue(seller);
    getRepresentative().setValue(representative);
    getAddress().setValue(address);
    getSellingPrice().setValue(sellingPrice);
    getAdditionalSurcharges().setValue(additionalSurcharges);
    getPaymentPeriodAndMethod().setValue(paymentPeriodAndMethod);
    getDeliveryTime().setValue(deliveryTime);
    getReturns().setValue(returns);
    getCancellation().setValue(cancellation);
}

}

public interface Payment {}

```

```

private static class StripeData implements Payment {
    static final String PAYMENT_SYSTEM = "Stripe";
    boolean isUSCarAssociationMemberCars;

    private StripeData(boolean isUSCarAssociationMemberCars) {
        this.isUSCarAssociationMemberCars = isUSCarAssociationMemberCars;
    }
}

private static class StripeData implements Payment {
    static final String PAYMENT_SYSTEM = "STRIPE";
    String tcicarrentalRentCommission;

    private StripeData(String tcicarrentalRentCommission) {
        this.tcicarrentalRentCommission = tcicarrentalRentCommission;
    }
}

private static class StripeData implements Payment {
    static final String PAYMENT_SYSTEM = "STRIPE";
    String tcicarrentalRentCommission;
    LegalInformationJpData legalInformationJpData;
    LegalInformationEnData legalInformationEnData;

    public StripeData(String tcicarrentalRentCommission,
        LegalInformationJpData legalInformationJpData,
        LegalInformationEnData legalInformationEnData) {

        this.tcicarrentalRentCommission = tcicarrentalRentCommission;
        this.legalInformationJpData = legalInformationJpData;
        this.legalInformationEnData = legalInformationEnData;
    }
}

private static class LegalInformationEnData {
    String seller;
    String representative;
    String address;
    String sellingPrice;
    String additionalSurcharges;
    String paymentPeriodAndMethod;
    String deliveryTime;
    String returns;
    String cancellation;

    public LegalInformationEnData(String seller, String representative,
        String address, String sellingPrice,
        String additionalSurcharges, String
        paymentPeriodAndMethod, String deliveryTime,
        String returns, String cancellation) {

        this.seller = seller;
        this.representative = representative;
        this.address = address;
        this.sellingPrice = sellingPrice;
        this.additionalSurcharges = additionalSurcharges;
        this.paymentPeriodAndMethod = paymentPeriodAndMethod;
        this.deliveryTime = deliveryTime;
        this.returns = returns;
        this.cancellation = cancellation;
    }
}

```

```
}  
  
private static class LegalInformationJpData extends LegalInformationEnData {  
    String phoneNumber;  
    String email;  
  
    public LegalInformationJpData(String seller, String representative,  
String address, String phoneNumber, String email, String sellingPrice, String  
additionalSurcharges, String paymentPeriodAndMethod, String  
deliveryTime, String returns, String cancellation) {  
  
        super(seller, representative, address, sellingPrice,  
additionalSurcharges, paymentPeriodAndMethod,  
                deliveryTime, returns, cancellation);  
        this.phoneNumber = phoneNumber;  
        this.email = email;  
    }  
}  
}
```

Кафедра КБПЗ – 2021 рік