Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра кібербезпеки
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(освітній ступінь (освітньо-кваліфікаційний рівень))

на тему: Розробка інтернет-магазину спортивних товарів “Travelua” та його захист

Виконав: студен IV курсу, групи СБс-42

спеціальності (напряму підготовки) 125 Кібербезпека

(шифр і назва спеціальності (напряму підготовки))

_____ Кулеба П.М.
(підпис) (прізвище та ініціали)

Керівник _____ Скоренький Ю. Л.
(підпис) (прізвище та ініціали)

Нормоконтроль _____ _____
(підпис) (прізвище та ініціали)

Завідувач кафедри _____ Загородна Н.В
(підпис) (прізвище та ініціали)

Рецензент _____ _____
(підпис) (прізвище та ініціали)

м. Тернопіль – 2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра кібербезпеки

(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В
(підпис) (прізвище та ініціали)
« » _____ 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр

(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека

(шифр і назва спеціальності)

студенту Кулебі Павлу Миколайовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інтернет-магазину спортивних товарів "Travelua" та його захист

Керівник роботи Скоренький Юрій Любомирович к.ф.-м.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 16 » 02 2021 року № 4/7-114

2. Термін подання студентом завершеної роботи 23.06.2022 р.

3. Вихідні дані до роботи Вимоги до функціоналу сайту інтернет-магазину, технічна документація

4. Зміст роботи (перелік питань, які потрібно розробити) Вступ. 1. Сучасні технології web-Розробки. 2. Особливості розробки сайту, який потребує захисту. 3. Пошук вразливостей та захист сайту. 4. Безпека життєдіяльності, основи охорони праці. Висновки. Перелік використаних джерел.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Титулка. 2. Актуальність теми, мета роботи. 3. Мови програмування, використані при розробці сайту. 4. Структура бази даних. 5. Основні вразливості сайтів. 6. Інтерфейс адмін-панелі та сторінка оплати. 7. Захист від DdoS атак сервісом Cloudflare. 8. Результат сканування вразливостей. 9. Заходи захисту сайту. 10. Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Пулька Ч. В. д.т.н, професор		

7. Дата видачі завдання 16.02.2021 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	16.02 – 19.02	*Виконано*
2.	Підбір джерел про загрози безпеці	19.02 – 02.03	*Виконано*
3.	Опрацювання джерел про загрози безпеці	03.03 – 02.03	*Виконано*
4.	Підбір джерел про існуючі засоби захисту сайтів	07.03 – 10.03	*Виконано*
5.	Опрацювання джерел про існуючі засоби захисту сайиів	10.03 – 16.03	*Виконано*
6.	Розробка сайту	16.03 – 01.04	*Виконано*
7.	Вибір хостингу	01.04 – 10.04	*Виконано*
8.	Оформлення розділу «Сучасні технології web-розробки»	10.04 – 16.04	*Виконано*
9.	Оформлення розділу «Особливості розробки сайту який потребує захисту»	16.04 – 25.04	*Виконано*
10.	Оформлення розділу «Пошук вразливостей та захист сайту»	25.04 – 05.05	*Виконано*
11.	Виконання завдання до підрозділу «Безпека життєдіяльності, основи охорони праці»	05.05 – 16.05	*Виконано*
12.	Оформлення кваліфікаційної роботи	16.05 – 22.05	*Виконано*
13.	Нормоконтроль	22.05 – 08.06	*Виконано*
14.	Перевірка на плагіат	08.06 – 10.06	*Виконано*
15.	Попередній захист кваліфікаційної роботи	10.06 – 19.06	*Виконано*
16.	Захист кваліфікаційної роботи	24.06	*Виконано*

Студент _____ *Кулеба П.М.*

(підпис) (прізвище та ініціали)

Керівник роботи _____ *Скоренький Ю.Л.*

(підпис) (прізвище та ініціали)

АНОТАЦІЯ

«Розробка інтернет-магазину спортивних товарів “Travelua” та його захист» // Кваліфікаційна робота освітнього рівня «Бакалавр» // Кулеба Павло Миколайович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп’ютерно-інформаційний систем і програмної інженерії, кафедра кібербезпеки, група СБс-42 // Тернопіль 2022 // С.53, рис.-16, табл – 4, бібліогр. – 13.

Ключові слова: WEB-САЙТ, СКАНЕР, ІНФОРМАЦІЙНА БЕЗПЕКА, БЕЗПЕКА ДАНИХ, DDOS АТАКА.

Кваліфікаційна робота містить 4 розділи:

В першому розділі зроблено аналітичний огляд існуючих рішень, вказано на доцільність роботи та визначено вимоги до програмної документації.

В другому розділі розроблено технічне завдання, розроблена структура сайту і web-сторінок, спроектовано логіку сайту, виконано тестування та розміщення сайту в мережі інтернет.

Третій розділ містить огляд основних вразливостей сайтів, виконано сканування вразливостей, захист від знайдених загроз.

В четвертому розділі розглянуті питання безпеки життєдіяльності та охорони праці.

Кваліфікаційна робота носить практично-орієнтований характер і як частину містить повноцінно-функціонуючий сайт, розміщений в мережі Інтернет.

ANNOTATION

"Sport internet shop “Travelua” development and its security //" Qualification work of educational level "Bachelor" // Kuleba Pavlo Mikolayovich "// Ternopil National Technical University named Ivan Pulyuy, Faculty of Computer Information Systems and Software Engineering, Department cybersecurity, SBs-42 group // Ternopil 2022 // C.53, fig.- 16, table - 4, bibliogr. - 13.

Keywords: WEBSITE, SCANNER, INFORMATION SECURITY, DATA SECURITY, DDOS ATTACK.

Qualification work contains 4 sections:

The first section summarizes the analytical review of existing documents, indicates the feasibility of the work and defines the requirements for program documentation.

In the second section the technical task is developed, the structure of the site and web-pages is developed, the logic of the site is designed, the testing and placement of the site on the Internet is performed.

The third section contains an overview of the main vulnerabilities of the sites, performed scanning of vulnerabilities, protection against detected threats.

The fourth section deals with issues of life safety and labor protection.

Qualification work is practice-oriented and as part of a full-featured site hosted on the Internet.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

API – Application Programming Interface

CMS – Content Management System

DDoS – Distributed Denial-of-service

FTP – File Transfer Protocol

PHP – Hypertext Preprocessor

SEO – Search Engine Optimization

HTML – HyperText Markup Language

HTTP – HyperText Transfer Protocol

HTTPS – Hyper Text Transfer Protocol Secure

IP – Internet Protocol

SSL – Secure Sockets Layer

SQL – Structured Query Language

URL - Uniform Resource Locator

XSS – Cross-Site Scripting

XML – Extensible Markup Language

ВСТУП

20 грудня 1990 року Тім Бернерс-Лі ввів першу в світі веб-сторінку в Інтернеті, що дало початок WWW. Вона народилася як система обміну даними між 10 000 вчених, які працювали в ЦЕРН. У ньому не було кольорів, зображень чи відео. Там також не було графіки чи анімації, лише текст, гіпертекст і набір меню.

Ось так воно і є. Перший у світі веб-сайт був створений у грудні 1990 року, але лише в 1993 році цю систему лібералізували, дозволяючи кожному, хто хотів створити свій власний веб-сайт.

Пізніше, в 1994 році, Бернерс-Лі створив консорціум World Wide Web Consortium (W3C), щоб підтримувати загальні стандарти роботи мережі. А в 1998 році він подумав про процес, який допоміг йому створити його, сказавши: «Якщо ви думаєте, що гіпертекстовий серфінг — це чудово, це тому, що ви ніколи не намагалися його написати».

На даний момент сайт може служити описом вашого підприємства, фірми, бути вашим портфоліо. Також існують файлообмінники, інтернет-магазини, соціальні мережі, онлайн-сервіси. Сайт допоможе поширювати інформацію, вчасно її оновляти, таким чином відвідувачі сайту завжди матимуть актуальну інформацію. Лише за наявності власного сайту організація не скована і може розповсюджувати будь-яку небхідну її інформацію на власну онлайн аудиторію.

Під час написання дипломного проекту будуть використанні такі мови програмування як HTML, CSS, JS та PHP.

JavaScript — це мова програмування або сценаріїв, яка дозволяє вам реалізовувати складні функції на веб-сторінках, щоразу, коли веб-сторінка робить щось більше, ніж просто сидить і відображає статичну інформацію для вас, вона відображає своєчасні оновлення вмісту, інтерактивні карти, 2D/3D.

Ще одна мова програмування яка використовувалася при розробці сайту є PHP (Personal Home Page Tools) — мова програмування, генерує HTML-сторінки на стороні сервера. PHP являється найпорулярнішою мовою, що

використовуються для веб-розробок. Більшість хостинг-провайдерів підтримують PHP.

1 СУЧАСНІ ТЕХНОЛОГІЇ WEB-РОЗРОБКИ

1.1 Аналітичний огляд існуючих рішень

Згідно з офіційним визначенням, інтернет-магазин є невід’ємною частиною процесу, відомого як інтернет-магазин. Інтернет-магазини – це процес, у якому відвідувачі можуть придбати певні продукти чи послуги, пропоновані в інтернет-магазині.

У системі онлайн-покупок або інтернет-магазину існують три найпоширеніших способи ведення бізнесу: B2C (скорочено від бізнесу до споживача), який передбачає, що процес інтернет-магазину відбувається між покупцем і підприємцем, тобто виробником або постачальник послуг, потім B2B (скорочено від бізнесу до бізнесу), в якому процес онлайн-торгівлі відбувається між двома підприємцями, а метод B2B2C розроблений як свого роду комбінація цих двох методів онлайн-торгівлі, що скорочено від англійських термінів business до бізнес клієнту, що означає, що онлайн-торгівля відбувається або між двома підприємцями, або між покупцем і підприємцем за умови, що в цьому способі торгівлі між ними також є посередник.

Саме цей третій спосіб також є найбільш поширеним у сучасних інтернет-магазинах. Справа в тому, що інтернет-торгівля значно полегшує покупку певних товарів і послуг. Однак не варто забувати, що це не пряма торгівля, є численні недоліки, які найчастіше стосуються питання якості продукції, що пропонується в інтернет-магазині. Зокрема, це стосується, наприклад, покупки супутніх товарів, пов’язаних з одягом, оскільки дуже часто зображення товару, яке користувач може побачити в Інтернеті, не повністю відповідає продукту. З цієї причини багато посередників у сфері інтернет-торгівлі на своїх сайтах також вказують можливість повернення товару. Однак це лише одна з можливих незручностей, з якими можуть зіткнутися клієнти, які користуються послугами інтернет-магазину.

У більшості інтернет-магазинів оплату можна здійснити всіма платіжними картками. Але багато в чому це стосується інтернет-магазинів, які

працюють на внутрішньому ринку. Для тих, хто є світовим лідером у цьому виді торгівлі, зазвичай необхідно мати спеціальну картку для онлайн-розрахунків, що значно ускладнює користування послугами таких інтернет-магазинів вітчизняним користувачам.

1.2 Вимоги до програмної документації

Документація до програмного забезпечення — це будь-яка документація, створена, щоб допомогти користувачам або розробникам зрозуміти деякі функції та функції програмного забезпечення. Цей тип технічної документації складається з письмових навчальних посібників, відео, посібників користувача та навчальних посібників, які мають на меті допомогти користувачам зрозуміти особливості, дії та функціональність програмного забезпечення.

Програмна документація має дві цільові аудиторії: інженерів-програмістів і кінцевих користувачів продукту. У розробці програмного забезпечення документація описує матеріали та документи, які допомагають інженерам зрозуміти дизайн, код і впровадження продукту. Ця документація дозволяє розробникам розуміти, оновлювати та налаштовувати програмне забезпечення зсередини. Для кінцевих користувачів документація відноситься до простого набору ресурсів, які пояснюють, як налаштувати та використовувати програмне забезпечення.

Після закінчення розробки даного веб-сайту необхідно підготувати наступну документацію:

- інструкція з розміщення сайту в Інтернеті;
- інструкція з обслуговування та наповнення сайту;
- опис основних можливостей даного веб-сайту;
- причини і усунення можливих збоїв в роботі.

1.3 Стадії та етапи розробки

Перший етап побудови інтернет-магазину є важливим завданням, що стоїть перед підприємцем. Він повинен вирішити, що продаватиме, і чи підходить даний товар для електронної торгівлі. На цьому етапі відбувається оцінка конкурентів: аналіз сайтів, які пропонують аналогічні товари/послуги.

На другому етапі підприємець повинен визначити, які функції повинен мати майбутній інтернет-магазин.

Третій етап – це розробка технічного завдання, створення веб-сайту. Цей процес здійснюють фахівці в галузі ІТ, добре обізнані про специфіку діяльності компанії. Перший крок розробки є визначення структури веб-сайту, дизайну, принципів роботи та розташування інформації. Тут же підбирається необхідне ПЗ. Після закінчення складання технічного завдання для створення веб-сайту інформаційний та програмний супровід може вести саме підприємство.

На наступному етапі вирішується питання розміщення сайту в Інтернеті. Існує кілька варіантів:

- на своєму сервері. При цьому він або знаходиться в комп'ютерній мережі провайдера за відповідну абонентську плату, або підключається до провайдера виділеної лінії;
- на устаткуванні провайдера (віртуальний сервер). І тут у провайдера орендується дисковий простір (хостинг). Цей варіант найменш витратний і підходить для проектів не пов'язаних з конфіденційною фінансовою інформацією (до сервера матимуть доступ співробітники провайдера) і не потребують використання специфічного програмного забезпечення, нестандартних програмно-апаратних конфігурацій.

1.4 Тестування сайту

Веб-тестування — це тип тестування програмного забезпечення, яке використовується для виявлення помилок у веб-сайтах і веб-додатках. Веб-програми ретельно перевіряються, перш ніж вони стануть доступними в

Інтернеті. Перш ніж будь-яка програмна служба або програма стане доступною для кінцевих користувачів, ви повинні перевірити їх на наявність помилок. Тестування веб-сайтів дозволяє командам розробників переконатися, що веб-система працює ефективно та забезпечує найкращий користувальницький досвід.

Існують різні способи тестування, проте необхідно памятати про стратегію і процес тестування. Від вибраної стратегії залежить послідосність дій.

Найчастіше застосовують такі види тестування для веб-сайтів:

- Тестування функціональності
- Тестування зручності використання
- Тестування на продуктивність

Розглянемо докладніше вище зазначені види тестування:

Говорячи про функціональне тестування мова йде про перевірку, чи все на сайті працює коректно. Наприклад, форми зворотнього зв'язку, відгуки, замовлення в інтернет-магазині, підписка на новини, розрахунок вартості, виклик майстра, все, що було задумано для взаємодії з потенційними клієнтами вашої компанії, має бути справно, завжди, 24/7. Рекомендується проводити функціональне тестування сайту регулярно, особливо якщо на сайті є складний функціонал, інтеграції з іншими системами, такими як складські програми, служби доставки, CRM і так далі.

Тестування зручності використання сайту. Кінцева ціль тестування зручності сайту – це максимально зрозумілий дизайн та структура яка не тільки не заваджатиме користувачу в виборі товарів, а і допоможе йому в цьому.

Тестування продуктивності. Визначення тестування продуктивності можна резюмувати як процес тестування системи, що навантажується для виявлення вузьких місць у продуктивності. У парасольці тестування продуктивності є підмножини тестування продуктивності, такі як тестування навантаження, стрес-тестування, тестування на витривалість, тестування на сплеск, тестування гучності та тестування масштабованості. Тестування навантаження та стрес-тестування, як правило, є найбільш популярними,

відомими типами тестування продуктивності, але кожен тип тестування продуктивності встановлює для виявлення та вирішення конкретних проблем, пов'язаних із продуктивністю.

2 ОСОБЛИВОСТІ РОЗРОБКИ САЙТУ ЯКИЙ ПОТРЕБУЄ ЗАХИСТУ

2.1 Технічне завдання

2.1.1 Найменування та область застосування

Назва веб-сайту – «Travelua».

Область застосування сайту - продаж туристичного спорядження.

2.1.2 Призначення розробки

Експлуатаційне призначення – надання інтернет користувачу місця для перегляду інформації про товар та відгуків до нього від інших користувачів, можливості купити та оплатити покупку онлайн за допомогою банківської карти.

Функціональне призначення – для виконання поставлених задач використано мову PHP та JavaScript, як найбільш прості в реалізації даної задачі. У зв’язку з веб-сервером Apache, дана програма може використовуватися в різних операційних системах і мережах любого типу.

2.1.3 Вимоги до функціоналу веб-сайту

Даний веб - сайт повинен бути адаптивним для всіх популярних платформ ресурсом, з привабливим дизайном та зрозумілою навігацією для користувачів.

Для відвідувача сайту має бути передбачений наступний функціонал:

- перегляд відкритої інформації на сайті;
- можливість реєстрації на сайті;
- залишати відгуки після авторизації на сайті;
- можливість оформлення замовлення після авторизації на сайті;
- можливість оплати товару онлайн.

Для адміністратора сайту має бути передбачений наступний функціонал:

- можливість додавати товар у магазин;
- можливість редагувати товар попередньо доданий у магазин;
- можливість видалення відгуків написаних користувачами;
- можливість видалення товару з магазину.

2.1.4 Техніко-економічні показники

Проект розроблявся на базі безкоштовного програмного забезпечення з відкритим кодом, затрати на реалізацію даного програмного забезпечення спрямовуються на оплату праці, електроенергію, оплату доменного імені та амортизацію обладнання.

2.2 Розробка структури сайту і web-сторінок

Структура веб-сайту (або архітектура веб-сайту) — це, по суті, те, як ви організуєте навігацію веб-сайтом і розмістите веб-сторінки на своєму сайті. Найкраща структура веб-сайту для SEO включає такі елементи, як чиста (або легка для виконання) панель навігації, внутрішні посилання, які слідують логічні шляхи для відвідувачів по всьому веб-сайту та можливість фільтрувати інформацію на додаткові підмножини або сторінки. Незалежно від того, де на вашому веб-сайті малого бізнесу може перебувати відвідувач, їм завжди повинно бути легко знайти речі на вашому веб-сайті.

Основна мета успішної структури SEO веб-сайту — зробити всю інформацію легко доступною для користувачів, щоб вони могли знайти її та вжити заходів якомога швидше. Заходь і виходь! Це означає, що вам потрібно пам’ятати про досвід користувачів, створюючи кожну сторінку на своєму веб-сайті. Ви завжди повинні ставити пріоритет дизайну та структури веб-сайту, переконавшись, що все добре розроблено, плавно й оптимізовано від навігації вашої домашньої сторінки до кнопок із закликом до дії.

Даний web-сайт містить наступні web-сторінки:

- Головна – для першого входу на сайт, та перегляду основної інформації

про сайт та інше;

- Контакти – для перегляду контактів та для зворотнього зв’язку з адміністратором сайту;

- Адмін-панель – для керування сайтом та його вмістом;

- Доставка – для перегляду способів доставки товару;

- Оплата – для перегляду інформації про способи оплати;

- Гарантія – для перегляду інформації про гарантію;

- Сторінка категорії товару – для перегляду категорії товарів;

- Сторінка товару – для перегляду інформації про товар;

- Сторінка корзини – для перегляду товарів доданих в корзину;

- Сторінка авторизації – для авторизації на сайті;

- Сторінка реєстрації – для реєстрації на сайті.

Користувачі, які мають права адміністратора можуть керувати даним сайтом та виконувати увесь допустимий функціонал.

2.3 Створення та верстка сторінок сайту

Для верстки сторінок сайту було використано наступний стек веб-технології:

- HTML (Hyper text Markup Language) розроблена для опису web-сторінки, яка зберігається з розширенням *.htm або *.html у виді текстового файлу;

- CSS це мова, яка використовується задання параметрів зовнішнього вигляду web-сторінок, написаних мовами розмітки даних таких як, HTML.

Кожна сторінка повинна мати шапку сайту. Шапка сайту – це сама верхня частина сайту, яка не змінюється при перегляді інших сторінок (header). На шапці завжди міститься логотип та навігаційне меню (див. рис. 2.1).

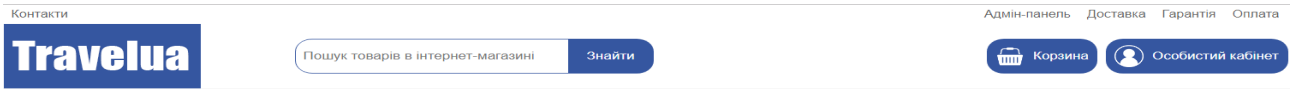

Рисунок 2.1 – Шапка web-сайту

Головна сторінка містить автомартичний слайдер з рекламними банерами, блок з категоріями товарів представлених на сайті, (див. рис. 2.2).

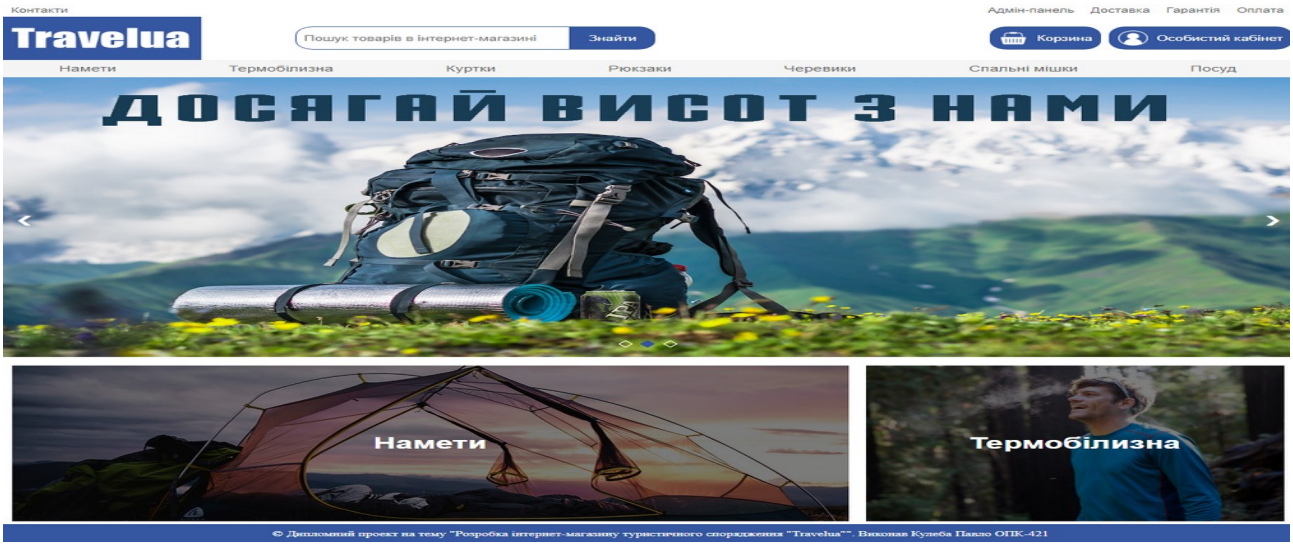

Рисунок 2.2 – Головна сторінка

Також повинен бути footer сайту, у футері міститься інформація про те ким був розроблений проект (див. рис. 2.3).

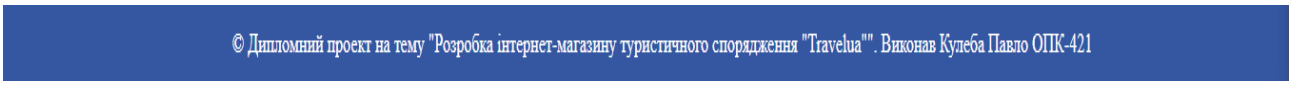

Рисунок 2.3 – Footer сайту

Сторінка з контактами містить всі необхідні контакти для зв’язку з адміністрацією, (див. рис. 2.4).

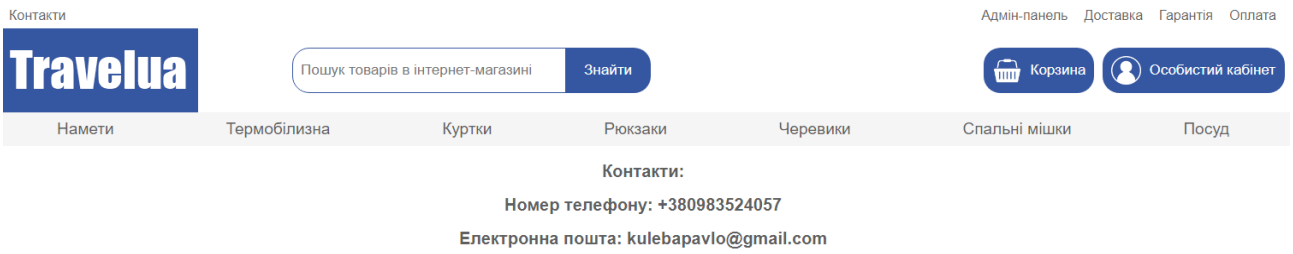

Рисунок 2.4 – Сторінка з контактами

Сторінка адмін-панелі містить кнопки для видалення товару, додавання товару, редагування товару та видалення коментарів, а також кнопку для переходу на головну сторінку та можливість сортування товарів по категоріях (див. рис. 2.5).

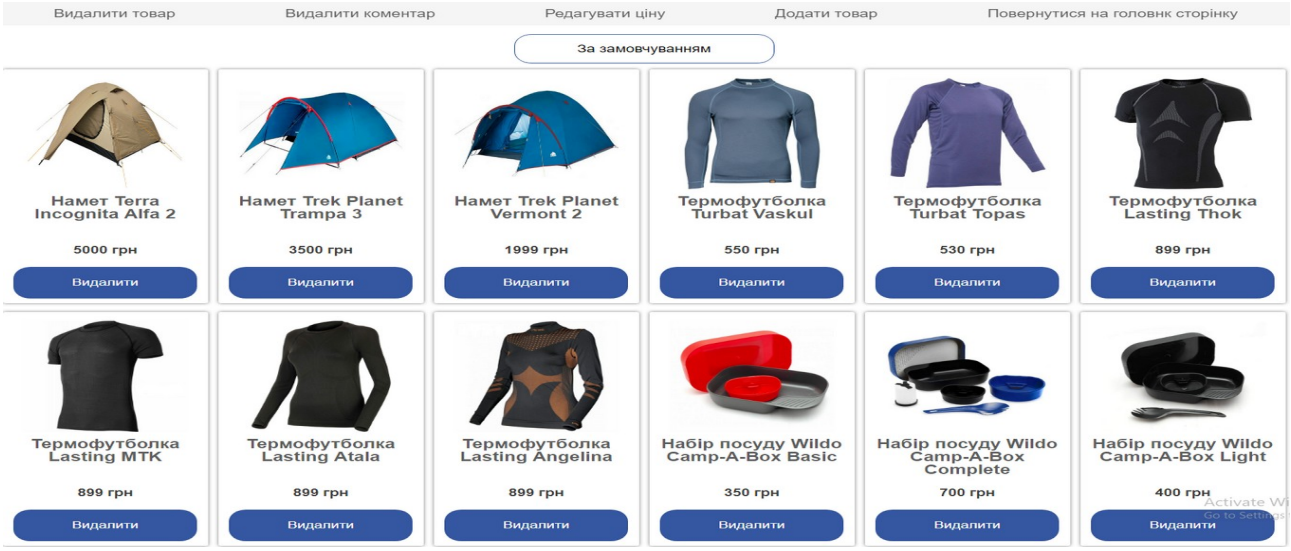

Рисунок 2.5 – Сторінка адмін-панелі

Сторінка «Оплата» інформує користувачів про способи оплати на сайті (див. рис. 2.6).

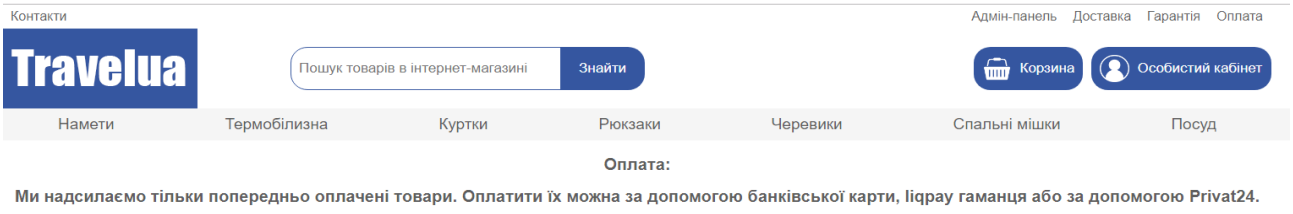

Рисунок 2.6 – Сторінка «Оплата»

Сторінка «Доставка» містить інформацію про способи доставки товарів (див. рис. 2.7).

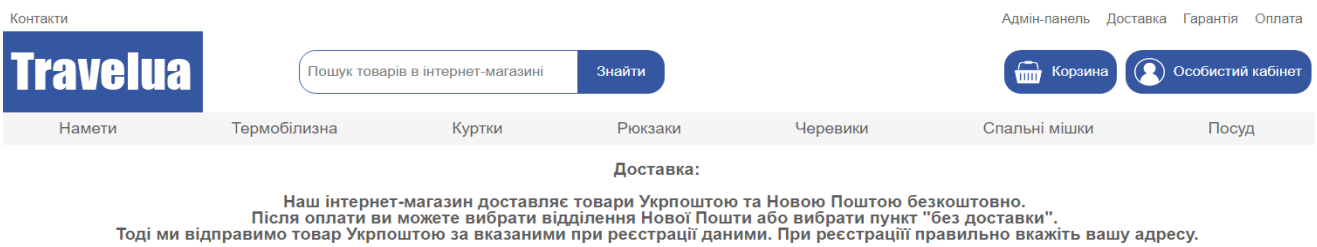

Рисунок 2.7 – Сторінка «Доставка»

Сторінка «Гарантія» містить інформацію про гарантію, яка надається на товар (див. рис. 2.8).

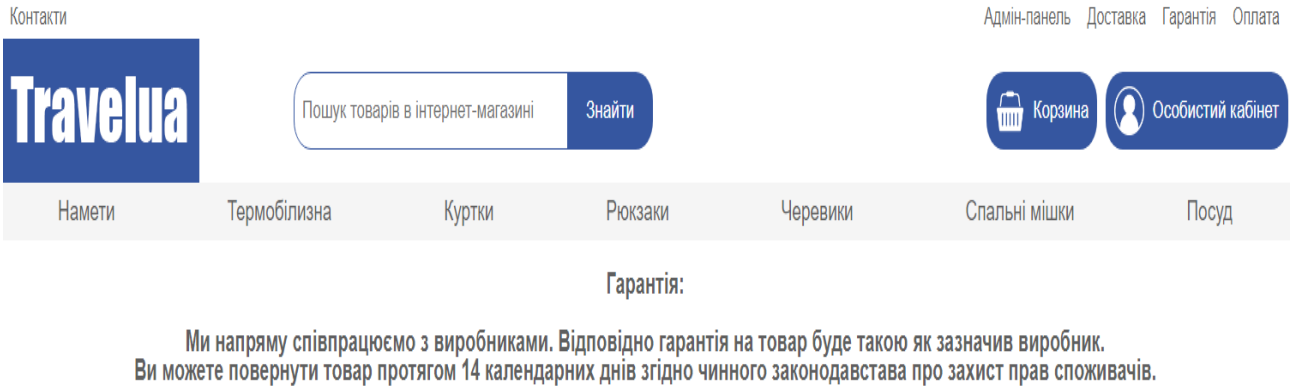

Рисунок 2.8 – Сторінка «Гарантія»

Сторінка категорії товару містить певну категорію товарів та можливість сортувати їх за ціною, популярністю та новизною (див. рис. 2.9).

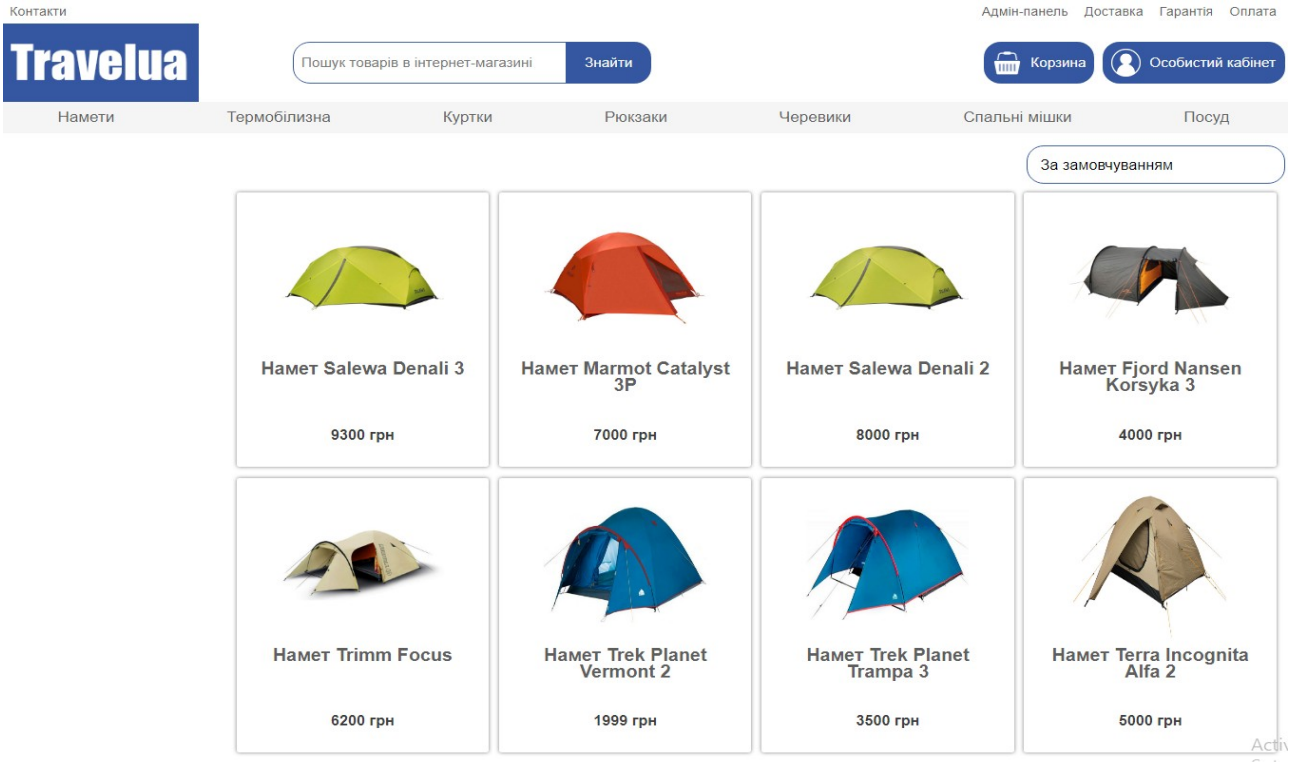

Рисунок 2.9 – Сторінка категорії товару

Сторінка товару містить фото, опис, ціну та назву товару, а також блок з відгуками до нього (див. рис. 2.10).

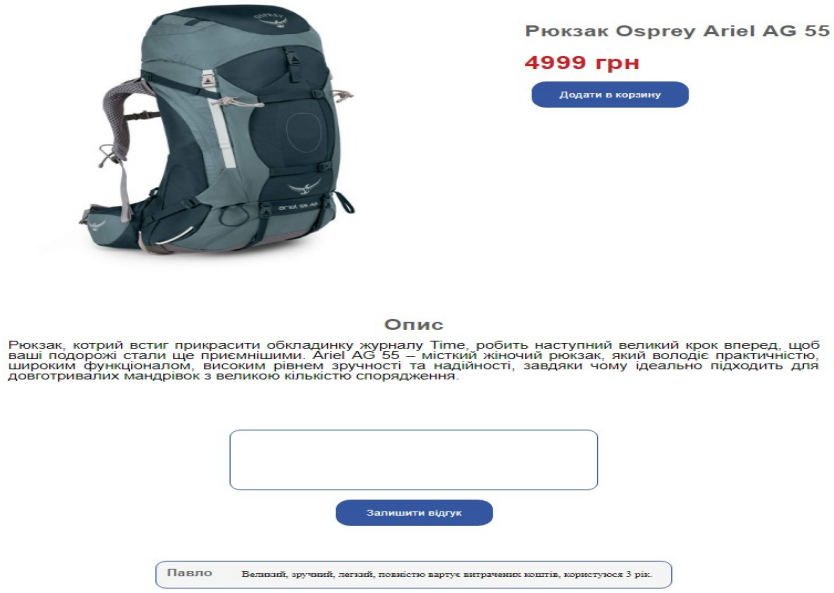

Рисунок 2.10 – Сторінка товару

Сторінка корзини містить товари, які користувач помістив в корзину, а саме фото, назву та ціну товару, а також показує загальну суму доданих в корзину товарів (див. рис. 2.11).

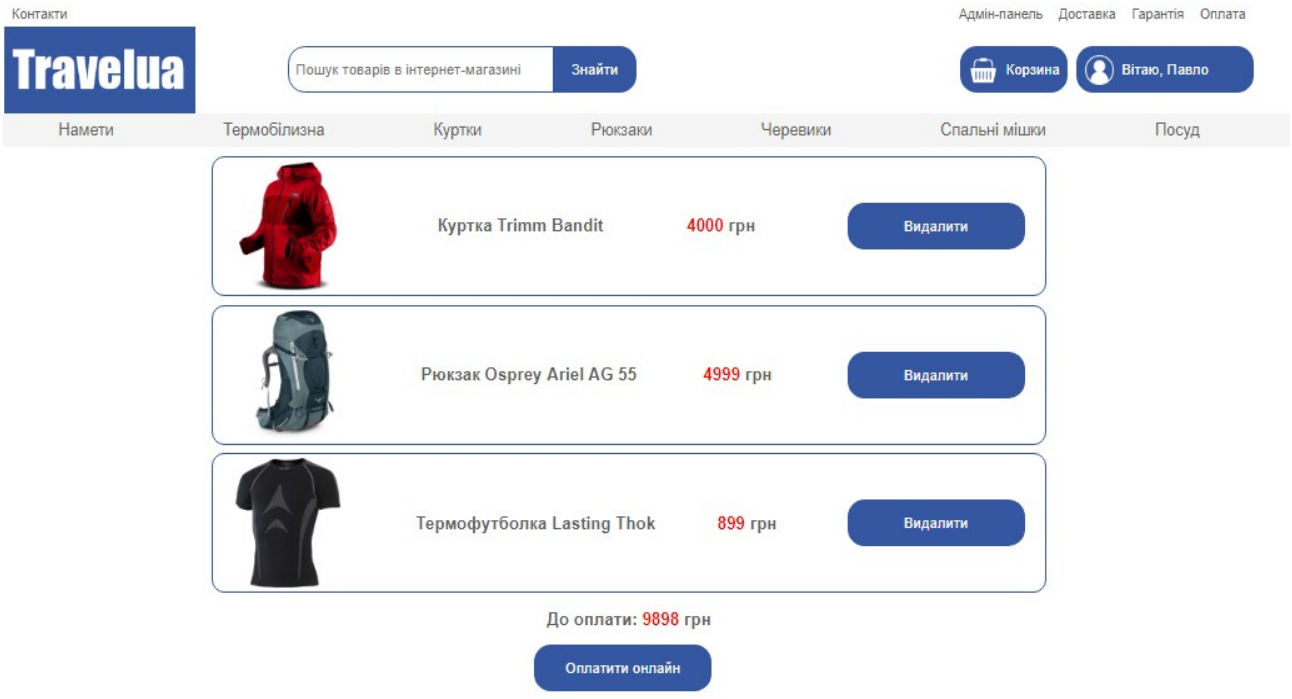

Рисунок 2.11 – Сторінка корзини

Сторінка авторизації містить поля для авторизації користувачем на сайті (див. рис. 2.12).

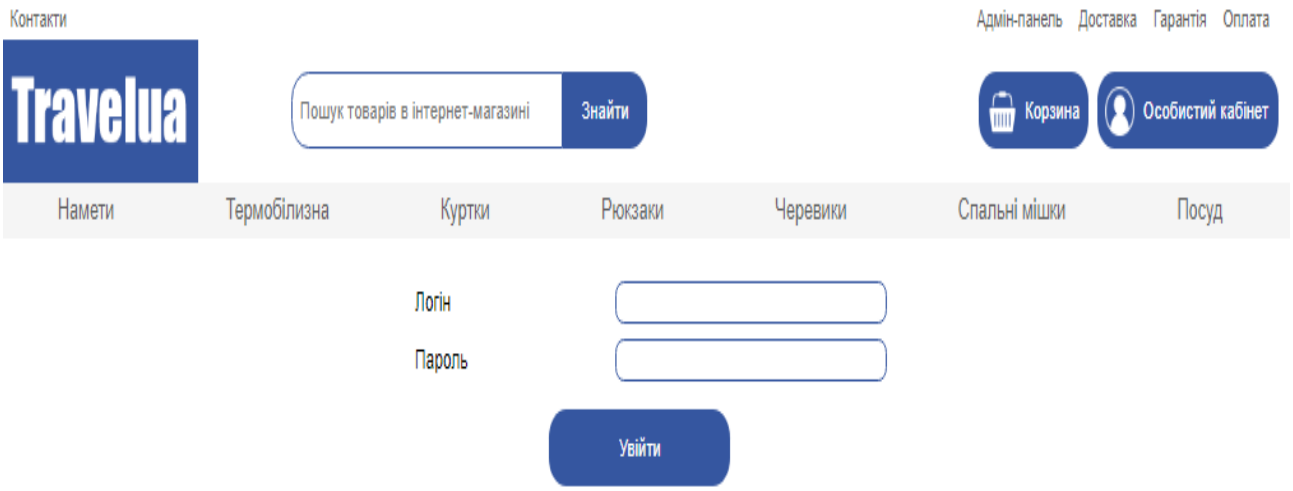

Рисунок 2.12 – Сторінка авторизації

Сторінка реєстрації містить поля для реєстраціїї користувачем на сайті (див. рис. 2.13).

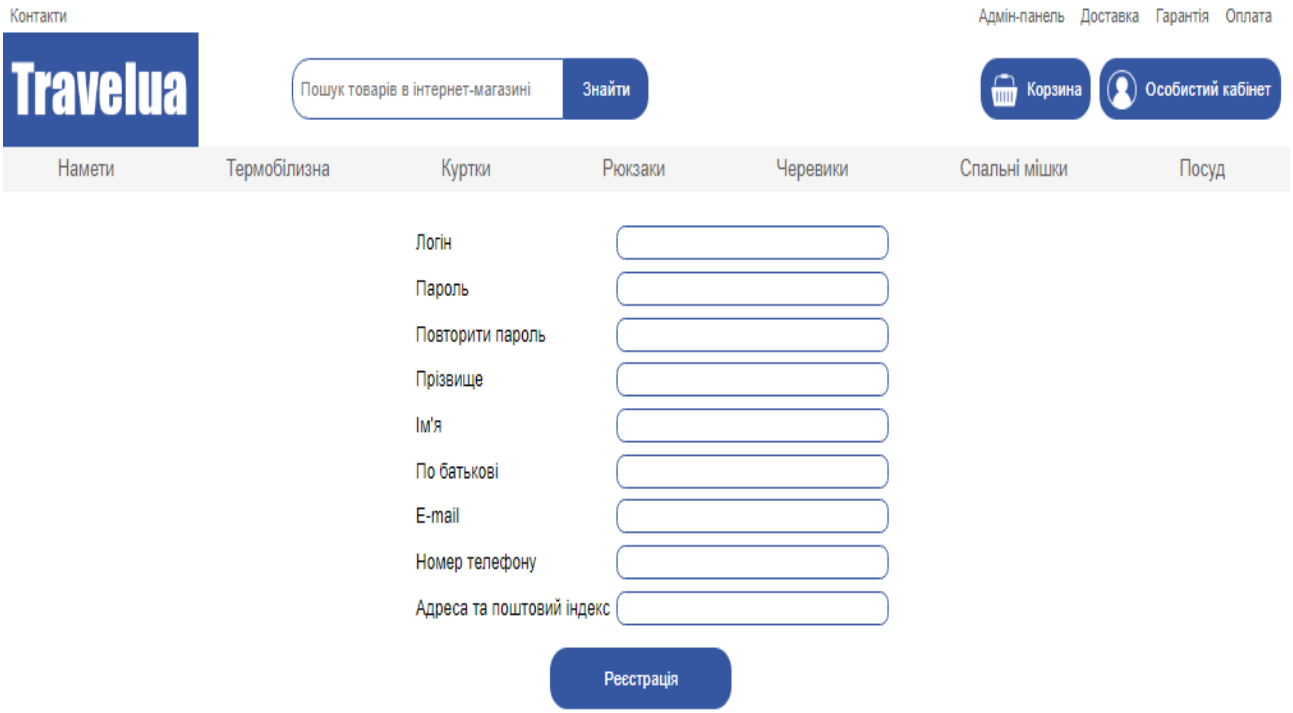

Рисунок 2.13 – Сторінка реєстрації

2.4 Розробка структури бази даних сайту

MySQL — це система управління базами даних з подвійною ліцензією. З одного боку, це відкритий код, а з іншого — комерційна версія, керована компанією Oracle. На даний момент це найвідоміша і використовувана база даних з відкритим вихідним кодом у всьому світі.

Як і він, ми можемо знайти інших, таких як сам Oracle або Microsoft SQL Server. Усі вони мають однакову мету та використовуються в одному середовищі, яке є не що інше, як веб-розробка, і на даний момент вони найбільше використовуються для формування та полегшення спілкування між веб-сайтами та серверами.

Однією з головних особливостей MySQL є те, що він працює з реляційними базами даних, тобто використовує кілька таблиць, які взаємодіють одна з одною, щоб зберігати інформацію та правильно її організовувати. Незважаючи на його призначення та середовище, в якому вона використовується, слід зазначити, що це система, спочатку розроблена на C і C++, одній з найтрадиційніших і найстаріших мов програмування, які існують.

Завдяки постійним оновленням і перевагам на користь того, щоб бути вільним і ідеально модифікованим середовищем, MySQL заслужив свою позицію як одного з найбільш використовуваних у цифровому секторі. Переважна більшість програмістів, які працюють у веб-розробці, пройшли через використання цього інструменту через його можливості та переваги.

База даних – це структуроване сховище даних. Для додавання інформації, її обробки, або для доступу до інформації, яка зберігається в базах даних на комп'ютерах, використовуються системи управління базами даних такі як MySQL. Сервер баз даних MySQL достатньо швидкий, надійний та легкий в управлінні.

Система управління базами даних – програмне забезпечення, що надає можливість створювати, зберігати, оновлювати та проводити пошук інформації в базах даних з контролем доступу до даних .

Для даного web-сайту була створена база даних «u597913845_dread». В базі даних були розроблені наступні таблиці для зберігання даних про товари, користувачів та відгуків, наведено у таблиці 2.1.

Таблиця 2.1 – Таблиці бази даних

Ім’я таблиці	Призначення
table_products	Містить дані про njdfhb
users	Містить дані про користувачів
coment	Містить дані з коментарями

table_products – таблиця відповідає за список товару та інформацію про нього, наведено у таблиці 2.2.

Таблиця 2.2 – Структура таблиці table_products

Ім’я поля	Тип поля	Призначення
products_id	int(11)	Ідентифікатор товару
title	varchar(255)	Назва товару
price	int(11)	Ціна товару
brand	varchar(255)	Бренд товару
image	varchar(255)	Зображення товару
description	text	Опис товару
count	int(11)	Кількість переглядів товару
type_product	varchar(255)	Тип товару
datetime	datetime	Дата додавання товару

users – таблиця відповідає за зберігання інформації про користувачів, зберігає унікальний ідентифікатор користувача, дані про нього, мобільний номер та електронну пошту, а токож адресу користувача, наведено у таблиці 2.3.

Таблиця 2.3 – Структура таблиці users

Ім’я поля	Тип поля	Призначення
id	int(11)	Ідентифікатор користувача
login	varchar(255)	Логін користувача
pass	text	Пароль користувача
surename	varchar(255)	Прізвище користувача
name	varchar(255)	Ім’я користувача
secondname	varchar(255)	По-батькові користувача
email	varchar(255)	Email користувача
phone	varchar(255)	Номер телефону користувача
address	text	Адреса користувача

coment – таблиця відповідає за зберігання відгуків, ідентифікатора користувача, який залишив відгук та ідентифікатор відгука, наведено у таблиці 2.4.

Таблиця 2.4 – Структура таблиці coment

Ім’я поля	Тип поля	Призначення
coment_id	int(11)	Ідентифікатор відгука
product_id	int(11)	Ідентифікатор товару
user_name	varchar(255)	Ім’я користувача
coment_text	text	Текст відгука

Усі таблиці створені за допомогою SQL запитів в phpmyadmin.

PhpMyAdmin – це open source додаток написаний мовою програмування PHP для обслуговування бази даних MySQL.

2.5 Програмування сайту

2.5.1 Написання клієнтської частини

Для написання клієнтської частини web-сайту було розроблено наступні моделі та функції:

- modalWindow – модуль для відкриття модального вікна для автоизації або реєстрації:

1) function btn() – функція відкриття модального вікна;

2) function span() – закриває модальне вікно при натисканні на кнопку;

3) function window() – закриває модальне вікно при натисканні на фон модального вікна.

- slider – модуль для відображення слайдера на головні сторінці сайту, описує такі функції:

1) function plusSlides() – функція для переходу на наступний слайд, зображено в додатку А, робить нективним помередній слайд та змінює дані кнопок на слайді для самостійного переходу на наступний або попередній;

2) function currentSlide() – функція змінює стилі слайду після переходу на нього, зображено в додатку Б, щоб той відображався на екрані, також змінює стилі крапки яка віповідає цьому слайду;

3) function showSlides() – функція автоматично виконує перехіх на натупний у списку слайд кожних 5 секунд, зображено в додатку В, також змінює стилі крапкам які відповідають конкретним слайдам.

- cart – модуль корзини магазину, описує такі функції:

1) function addToCart() – поміщає товар обраний користувачем в масив корзини, зображено в додатку Г;

2) function showMyCart() – відображає на сторінці корзини товари додані користувачем;

3) function delFromCart () – видаляє товари з корзини.

Також на стороні клієнта реалізована система оплати банківською картою.

Цей модуль розроблений за допомогою безкоштовного LIQPAY API, зображено в додатку Д.

API — це абревіатура від Application Programming Interface. Завдяки цим правилам і специфікаціям програми, які ви використовуєте, можуть спілкуватися один з одним.

Простіше кажучи, концепція API має формальний характер і відповідає комп'ютерним функціям і протоколам, за допомогою яких розробники можуть створювати конкретні програми для баз даних, операційних систем, онлайн-платформ або соціальних мереж. Саме інтерфейс полегшує спілкування різних програм.

Його використання залежить від дозволів, які власник API надає стороннім розробникам. Ви їх не бачите, але вони є внутрішнім проводом, який з'єднується з кодом і правилами для роботи програми. Увійти в онлайн-ігри зі свого облікового запису Facebook або за допомогою програми для спільної роботи, яка надсилає сповіщення на ваш ПК, можна завдяки API.

Щоб зрозуміти це, уявіть, що API — це як портфель інструментів, деякі з них повні, а інші мають лише одну функцію. Їх використання необмежене, оскільки їх використовують державні установи, приватні компанії, місцеві організації та багато іншого.

На кожній сторінці з товаром автоматично генерується JSON стрічка з параметрами відповідного товару.

Приклад JSON стрічки товару: json_string = {"public_key":"i00000000","version":"3","action":"pay","amount":"3","currency":"UAH","description":"test","order_id":"000001"}.

Кожне поле має своє значення:

- public_key. Це поле особистого публічного ключа, який видається власнику інтернет магазину в особистому кабінеті LIQPAY;

- version. Версія API;

- action. В цьому полі визначається тип платежу, в даному випадку pay, тобто перерахунок коштів на рахунок магазину, також тут може бути блокування коштів на рахунку покупця, підписка на регулярний платіж або пожертвування;

- amount. Це поле відповідає за ціну товару;

- currency. Валюта платежу;

- description. Коментар до платежу;

- order_id. Унікальний ідентифікатор платежу.

Після створення JSON стрічки шифруємо її за допомогою стандартної функціїї мови PHP – base64encode. Після цього отримуємо стрічку data яка в подальшому буде використовуватися для надсилання запиту на сервер LIQPAY.

Отримавши стрічку data необхідно сформувати стрічку sign_string. Для її створення необхідно шляхом конкатенації обєднати два приватних ключі виданих в кабінеті LIQPAY, всередені яких буде стрічка data.

Конкатенація (об'єднання) — операція обєднання двох змінних в одну, найчастіше текстових. Наприклад, конкатенація слів «веб» і «сайт» дасть слово «вебсайтццц».

Приклад конкатенації: $sign_string = $private_key . $data . $private_key.

Після отримання sign_string її необхідно захешувати за допомогою функції sha1() після цього зашифрувати за допомогою base64encode(), після цих маніпуляцій отримуємо рядок signature. Він необхідний для захисту даних платежу. Створивши створити хеш рядка sign_string, розшифрувати його стає неможливо.

Маючи рядки data та signature ми надсилаємо їх в html-формі на сервер LIQPAY, якщо все зроблено правильно, відбудеться переадресація на сторінку оплати (див. рис. 2.13).

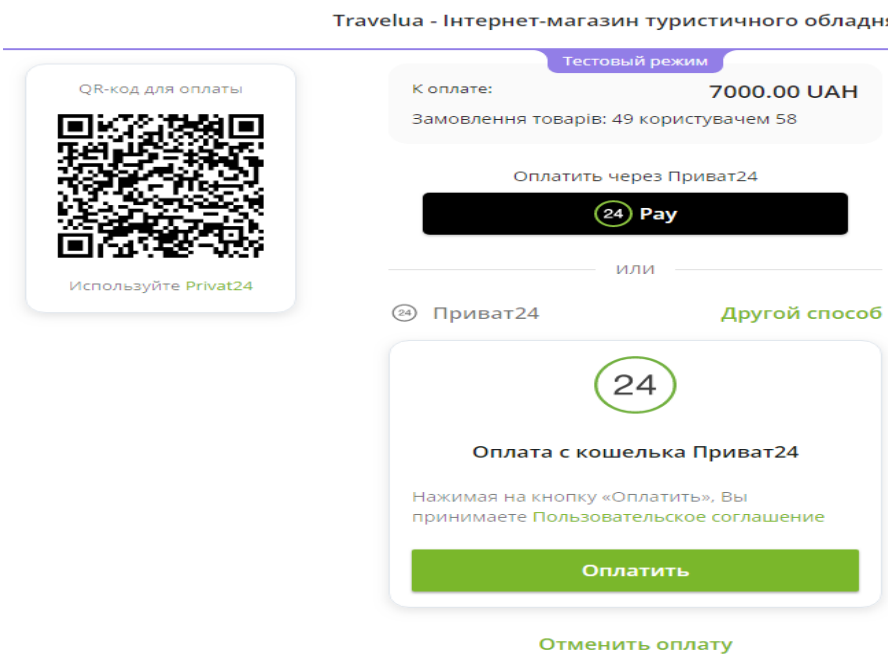

Рисунок 2.14 – Сторінка оплати

2.5.2 Написання admin-частини

Для написання адміністраторської частини web-сайту було використано мову програмування PHP та розроблено наступні моделі:

- db_connect – дана модуль призначений для підключення до бази даних, описує такі два методи:

1) $mysqli = new mysqli – встановлює підключення з базою даних;

2) if (mysqli_connect_errno()) – виконує перевірку підключення до бази даних.

- dellprod – модуль для видалення товарів, описує такі методи:

1) if(isset($_POST['products_id'])) – зчитує id товару;

2) $ressult = mysqli_query($mysqli, $dell) – видаляє товар із таблиці під певним id.

- dellcoment – модуль для видалення відгуків, , описує такі методи:

1) if(isset($_POST['coment_id'])) – зчитує id відгука;

2) $ressult = mysqli_query($mysqli, $dell) – видаляє відгук із таблиці під певним id.

- update – модуль для редагування ціни товару, описує такі методи:

1) if(isset($_POST['products_id']) && ($_POST['new_price'])) – зчитує id відгука та нову ціну товару;

2) $ressult = mysqli_query($mysqli, $update); – видаляє відгук із таблиці під певним id.

- add – модуль для додавання товару, описує такі методи:

1) if(isset($_POST['title']) && ($_POST['description')) – зчитує дані про новий товар;

2) $ressult = mysqli_query($mysqli, $add); – додає новий товар в базу даних.

Для захисту адмін-панелі від несакціонованого доступу було обрано захист авторизацією засобами веб-серверу Apache. Простота та надійність є основними перевагами цього способу.

Для захисту адмін-панелі необхідно в кореневій папці на хостингу де знаходяться файли та сторінки адмін-панелі створити файл .htaccess та прописати там наступні рядки з такими параметрами:

- AuthName. Рядок тексту, який виведеться при авторизації;

- AuthType. Тип аутентифікації. Basic або Digets. Вибрано Basic, так як не всі веб-переглядачі підтримують метод Digits;

- AuthUserFile. Шлях до файлу .htpasswd. В ньому записаний зашифрований логін та пароль адміністратора;

- Require valid-user. Рядок який дає доступ до дерективи тільки успішно авторизованим користувачам.

Також сайт захищений SSL сертифікатом. SSL (Secure Sockets Layer — рівень захищених сокетів) — криптографічний протокол, який забезпечує встановлення безпечного з'єднання між клієнтом і сервером.

3 ПОШУК ВРАЗЛИВОСТЕЙ ТА ЗАХИСТ САЙТУ

3.1 Основні вразливості сайтів

У пошуках сучасних методів та інструментів, що дозволяють нам хоч з якоюсь часткою впевненості стверджувати, що веб-сайт захищений від майбутніх хакерських атак (у тому, що вони будуть у всіх, ні в кого немає сумнівів? І якщо їх ще не було, то це лише питання часу), було знайдено рекомендації, які у цій статті розглянемо.

Цей невеликий, але важливий список конкретних дій, який слід робити кожному зі своїм веб-ресурсом, якщо репутація компанії, безпека веб-ресурсів та даних клієнтів – це не пусті слова для вас.

Можна виділити кілька основних способів захистити свій сайт:

- забезпечити захист від DDoS-атак;
- підключити SSL-сертифікат;
- використовувати надійний хостинг;
- використовувати SFTP замість FTP для передачі даних;
- застосовувати існуючі техніки захисту від SQL-ін'єкцій та XSS-атак;

Звичайно, у кожного пункту є своє «але» та ряд підпунктів, на яких слід загострити увагу. Також їх можна розділити на підгрупи виходячи з наступних міркувань: одні дії вимагають одноразового підключення, налаштування та рідкісних перевірок працездатності (налаштування хостингу та SSL-сертифіката), а інші мають на увазі постійні перевірки, оновлення та вимагають пильної уваги (все інше).

3.1.1 DDoS-атаки

DDoS – скорочення від словосполучення Distributed Denial of Service. Це атака на обчислювальну систему, що призводить до її відмови. Найчастіше вона виконується одночасно з безліччю комп'ютерів. Наслідки: перевантаження

сервера, уповільнення чи припинення роботи сайту. Що відбувається при Distributed Denial of Service? Навіщо хакери застосовують ці дії? Сервер, на якому знаходиться онлайн-ресурс, перевантажується запитами настільки, що не здатний їх обробити. В результаті сайт блокується і стає недоступним.

DDoS атака – це зловмисні дії з метою нашкодити конкуренту або викупити за припинення атаки. Найпоширеніші причини DDoS атак:

- знизити рейтинг онлайн-ресурсу в пошуковій системі;
- помститися власнику сайту;
- зробити онлайн-ресурс недоступним під час масового продажу (це можуть бути свята тощо);
- вимагання грошей за припинення атаки;
- незадоволений клієнт вирішив нашкодити компанії.

Якщо кібератака вже відбулася, то мінімізувати збитки від атаки хакера миттєво – неможливо. Втрати у разі неминучі. Чим раніше будуть вжиті захисні дії, тим меншими будуть збитки. Сайт можуть вивести з ладу, наприклад, під час сезонних розпродажів, пікового напливу клієнтів – наприклад, у розпал чорної п'ятниці або під Новий Рік. У такому разі власник ресурсу отримає збитки та удар по репутації. Щоб цього не сталося, потрібно подбати про захист ресурсу до настання такої ситуації. Є методи боротьби з хакерськими нападами, які допоможуть впоратися з атакою хоча б тимчасово. Після цього – необхідно звернутися до професійного сервісу (лікування та відновлення сайту), або до служби, яка надає послуги хостингу. Є й інші небезпеки DDoS-атак. Іноді кіберзлочинці за допомогою подібних дій намагаються відвернути увагу від інших дій. І поки ви будете боротися з несподіваною атакою хакера, у вас можуть викрасти конфіденційну інформацію, що знаходиться на тому ж самому сайті.

Захиститися від хакерів можна самостійно різними способами. Наприклад, настроювання конфігураційних файлів Apache, використання модуля mod_security для Apache. Діючим засобом є встановлення проксі-сервера nginx у зв'язці з Apache, що посилить безпеку системи. Nginx у цьому

випадку посилить Apache і забиратиме на себе обробку статичного контенту. Найпопулярніший спосіб захисту – використання сервісу Cloudflare. Сервіс, який допоможе приховати свою IP-адресу і, таким чином, стати недоступним для зловмисників. Є як безкоштовні тарифи, і платні. Безкоштовно можна захистити сайт не від усіх типів нападів. Якщо атака потужна, то потрібний буде платний тариф. Але встановлення Cloudflare потребує підключення спеціаліста, за роботу якого потрібно буде заплатити.

Збільшення швидкості завантаження сайту відбувається завдяки технології CDN. Cloudflare діє як посередник між клієнтом і сервером, використовуючи так звані «зворотні проксі» для створення дзеркальних копій і кешів веб-сайтів і, таким чином, забезпечує швидший, низьку затримку та більш безпечний доступ. Вони також можуть виявляти шкідливий трафік і обмежувати спам у мережі, серед інших заходів безпеки.

Це трапляється в рідкісних випадках, але інфраструктура може вийти з ладу, і саме тоді з’явиться помилка 502, через що веб-сайти, які залежать від CDN, не працюватимуть. У разі збою Cloudflare це може призвести до блокування мільйонів веб-сторінок. У таких ситуаціях ми мало що можемо зробити, окрім як терпляче чекати, поки вони самі вирішать проблему.

3.1.2 SSL-сертифікат

Сертифікат SSL – це цифровий сертифікат, який підтверджує автентифікацію веб-сайту та забезпечує зашифроване з’єднання. SSL означає Secure Sockets Layer, протокол безпеки, який створює зашифроване посилання між веб-сервером і веб-браузером. Компанії та організації повинні додавати сертифікати SSL на свої веб-сайти, щоб захистити онлайн-транзакції та зберегти конфіденційність та безпеку інформації клієнтів.

Коротше кажучи: SSL забезпечує безпеку інтернет-з’єднань і не дає зловмисникам читати або змінювати інформацію, передану між двома

системами. Коли ви бачите значок замка поруч із URL-адресою в адресному рядку, це означає, що SSL захищає веб-сайт, який ви відвідуєте.

З моменту свого заснування близько 25 років тому існувало кілька версій протоколу SSL, кожен з яких у певний момент зіткнувся з проблемами безпеки. Далі з’явилася оновлена та перейменована версія — TLS (Transport Layer Security), яка використовується й сьогодні. Однак ініціали SSL застрягли, тому нова версія протоколу як і раніше зазвичай називається старою назвою.

SSL працює, гарантуючи, що будь-які дані, передані між користувачами і веб-сайтами, або між двома системами, залишаються неможливими для читання. Він використовує алгоритми шифрування для сшифрування даних під час передачі, що не дозволяє хакерам прочитати їх, коли вони передаються через з’єднання. Ці дані містять потенційно конфіденційну інформацію, таку як імена, адреси, номери кредитних карток або інші фінансові дані.

Процес працює так: Браузер або сервер намагаються підключитися до веб-сайту (тобто веб-сервера), захищеного за допомогою SSL. Браузер або сервер запитують, щоб веб-сервер ідентифікував себе. У відповідь веб-сервер надсилає браузеру або серверу копію свого сертифіката SSL. Браузер або сервер перевіряє, чи довіряє він сертифікату SSL. Якщо це так, він сигналізує про це веб-серверу.

Потім веб-сервер повертає підтвердження з цифровим підписом для початку сеансу, зашифрованого SSL. Зашифровані дані обмінюються між браузером або сервером і веб-сервером. Цей процес іноді називають «рукостисканням SSL». Хоча це звучить як тривалий процес, він відбувається за мілісекунди.

3.1.3 Захист від SQL-ін'єкцій

Ін’єкція SQL є одним із найпоширеніших механізмів веб-атак, які використовуються зловмисниками для крадіжки конфіденційних даних з організацій. Хоча ін’єкція SQL може впливати на будь-яку програму, керовану

даними, яка використовує базу даних SQL, вона найчастіше використовується для атаки на веб-сайти.

SQL Injection — це техніка введення коду, яку хакери можуть використовувати для вставки шкідливих операторів SQL у поля введення для виконання базовою базою даних SQL. Ця техніка стала можливою через неправильне кодування вразливих веб-додатків. Ці недоліки виникають через те, що поля вводу, доступні для введення користувача, несподівано дозволяють операторам SQL проходити через і безпосередньо запитувати базу даних.

Зловмисники постійно перевіряють загальні веб-сайти в Інтернеті та університетські веб-сайти на предмет вразливостей ін'єкцій SQL. Вони використовують інструменти, які автоматизують виявлення недоліків SQL-ін'єкції, і намагаються використати SQL-ін'єкцію в першу чергу для фінансової вигоди (наприклад, крадіжки ідентифікаційної інформації, яка потім використовується для крадіжки особистих даних). Оскільки так багато сучасних програм керуються даними та доступні через Інтернет, уразливості SQL Injection широко поширені та легко використовуються.

Крім того, через поширеність інфраструктури спільної бази даних, недолік SQL Injection в одній програмі може призвести до компромісу інших програм, які використовують той самий екземпляр бази даних.

Захист від злому зводиться до базового правила "довіряй, але перевіряй". Необхідно перевіряти всі дані, такі як дати, числа, дані які користувач вводить у форми.

Більшість випадків ін’єкції SQL можна запобігти, використовуючи параметризовані запити (також відомі як підготовлені оператори) замість конкатенації рядків у запиті.

Наведений нижче код вразливий до ін’єкції SQL, оскільки введені користувачем дані об’єднуються безпосередньо в запит:

```
String query = "SELECT * FROM products WHERE category = '"+ input + "'";

Statement statement = connection.createStatement();
```

ResultSet resultSet = statement.executeQuery(query);

Цей код можна легко переписати таким чином, щоб уникнути втручання користувача в структуру запиту:

PreparedStatement statement = connection.prepareStatement("SELECT * FROM products WHERE category = ?");

statement.setString(1, input);

ResultSet resultSet = statement.executeQuery();

Параметризовані запити можна використовувати для будь-якої ситуації, коли ненадійне введення відображається як дані в запиті, включаючи WHERE речення та значення в операторі INSERT або UPDATE. Їх не можна використовувати для обробки ненадійного введення в інших частинах запиту, наприклад, назви таблиць, стовпців або ORDER BY речення. Функціональні можливості програми, які розміщують недовірені дані в цих частинах запиту, повинні використовувати інший підхід, наприклад, додавати дозволені значення введення в білий список або використовувати іншу логіку для забезпечення необхідної поведінки.

Щоб параметризований запит був ефективним для запобігання ін’єкції SQL, рядок, який використовується в запиті, завжди повинен бути жорстко закодованою константою і ніколи не повинен містити змінні дані з будь-якого походження. Не піддавайтеся спокусі вирішувати для кожного окремого випадку, чи є елемент даних довіреним, і продовжуйте використовувати конкатенацію рядків у запиті для випадків, які вважаються безпечними. Дуже легко зробити помилки щодо можливого походження даних або зміни в іншому коді порушують припущення про те, які дані зіпсовані

3.2 Пошук вразливостей за допомогою сканера Nikto

Для тестування сайту було обрано безкоштовний онлайн сканер варазливостей Nikto-онлайн.

Nikto — це сканер веб-сервера з відкритим вихідним кодом, який виконує комплексні тести веб-серверів для кількох елементів, включаючи понад 6700 потенційно небезпечних файлів/програм, перевіряє застарілі версії понад 1250 серверів та проблеми, пов’язані з версією, на понад 270 серверах. Він також перевіряє елементи конфігурації сервера, такі як наявність кількох індексних файлів, параметри HTTP-сервера, і намагатиметься ідентифікувати встановлені веб-сервери та програмне забезпечення. Елементи та плагіни сканування часто оновлюються і можуть оновлюватися автоматично.

Система складається з однієї сторінки, яка має наступний вигляд (див. рис. 3.1).

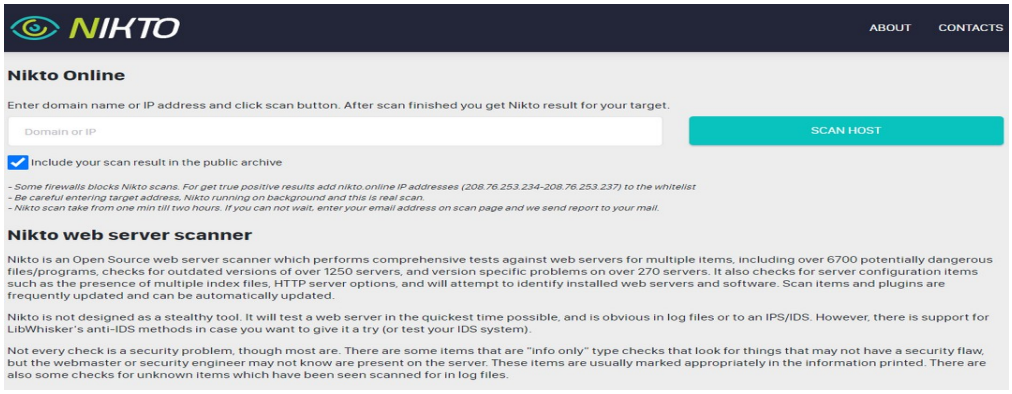

Рисунок 3.1 – Головна сторінка сайту Nikto

Для запуску сканування необхідно ввести в стрічку доменне ім’я сайту або його IP-адресу. Після чого почнеться сканування, яке триватиме від двох хвилин до двох годин. Для зручності та економії часу сервіс пропонує ввести email-адресу куди в подальшому буде відправлено звіт з результатами сакнування. Після завершення сканування отримуємо результат (див. рис. 3.2).

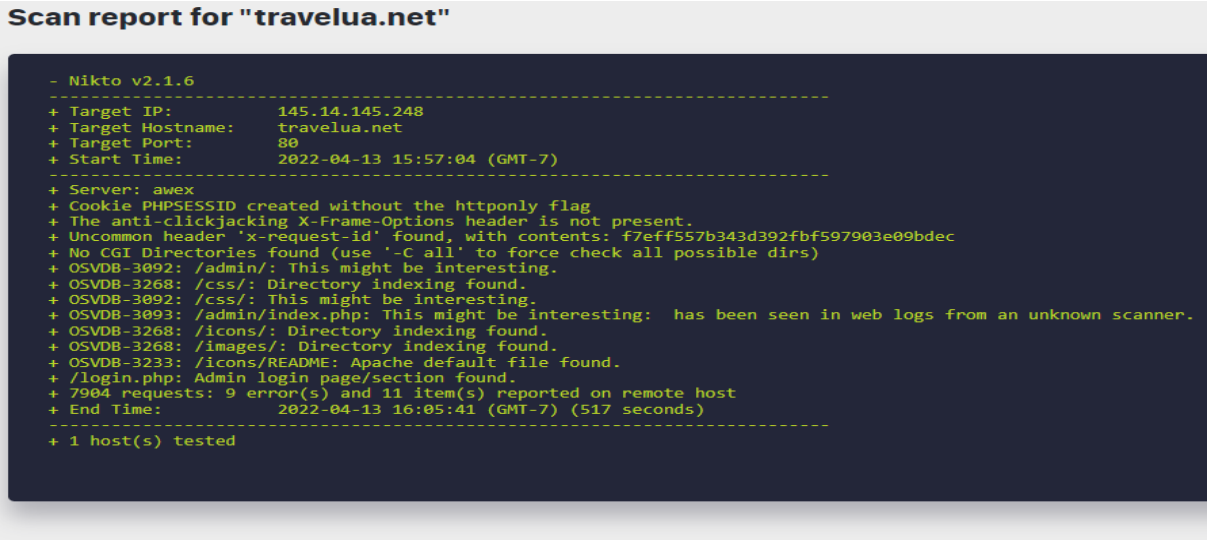

Рисунок 3.2 – Результати сканування

В результаті сканування було виявлено такі вразливості:

- Cookie PHPSESSID, створений без прапорця httponly;
- Заголовок X-Frame-Options проти клацання відсутній;
- Знайдено незвичайний заголовок "x-request-id" із вмістом: f7eff557b343d392fbf597903e09bdec;
- Не знайдено каталогів CGI;
- login.php: знайдено сторінку/розділ входу адміністратора.

Також знайдено вразливості з OSVDB. Open Sourced Vulnerability Database (OSVDB) - некомерційний проект, що представляє незалежну базу даних уразливостей, з відкритим вихідним кодом. Назва розшифровується приблизно як: "Відкрита база даних уразливостей". Основна його мета полягалє у відслідковуванні вразливостей та пошуку засобів для їх усунення, сервіс надає докладну та актуальну технічну інформацію для тих хто захищав мережеві системи від випадкових зловживань та систематичних атак, починаючи з домашніх користувачів та малих підприємств та закінчуючи транснаціональними корпораціями. Проект стимулє до більш відкритої співпраці між компаніями та окремими програмістами, що призводило до зниження витрат з пошуком та усуненням уразливостей у базах даних. Девіз OSVDB: "Все вразливе". Знайдено такі можливі вразливості:

- OSVDB-3092: Це вважається незначною вразливістю щодо розкриття інформації;
- OSVBD-3093: Було помічено у логах невідомий сканер;
- OSVBD-3268: Знайдено індексацію каталогу;
- OSVDB-3233: Знайдено файл Apache за замовчуванням.

Більшість з цих вразливостей не несуть загрози для безпеки сайту та служать як маркери для перевірки. Сканування вразливостей зайняло 8 хвилин.

3.3 Захист сайту від знайдених вразливостей

3.3.1 Файл cookie без позначки HttpOnly

Програмне забезпечення використовує файли cookie для зберігання конфіденційної інформації, але файл cookie не позначений прапорцем HttpOnly.

Прапор HttpOnly спрямовує сумісні браузери забороняти клієнтському сценарію доступ до файлів cookie. Включення прапора HttpOnly у заголовок відповіді Set-Cookie HTTP допомагає зменшити ризик, пов’язаний із міжсайтовими сценаріями (XSS), коли код сценарію зловмисника може спробувати прочитати вміст файлу cookie та вилучити отриману інформацію. Якщо все буде налаштовано правильно, браузери, які підтримують цей прапор, не розкриватимуть вміст файлу cookie третій стороні за допомогою клієнтського сценарію, що виконується через XSS.

Файл cookie HTTP – це невеликий фрагмент даних, який приписується певному веб-сайту та зберігається на комп’ютері користувача веб-браузером користувача. Ці дані можна використовувати для різних цілей, включаючи збереження інформації, введеної в поля форми, запис активності користувачів, а також для аутентифікації. Файли cookie, які використовуються для збереження або запису інформації, створеної користувачем, доступні та змінені за допомогою коду сценарію, вбудованого у веб-сторінку. У той час як файли cookie, які використовуються для аутентифікації, створюються сервером веб-сайту і надсилаються користувачеві для додавання до майбутніх запитів. Ці файли cookie для аутентифікації часто не призначені для доступу на веб-сторінці, надісланій користувачеві, а натомість вони повинні бути додані до майбутніх запитів для перевірки деталей автентифікації.

Якщо прапор HttpOnly не встановлено, конфіденційна інформація, що зберігається в файлі cookie, може бути відкрита для ненавмисних сторін.

В даному проекті файл cookie, про який йде мова, є файлом аутентифікації, тобто відсутність прапора HttpOnly може дозволити

зловмисникові вкрасти дані аутентифікації (наприклад, ідентифікатор сеансу) і прийняти особу користувача, отримати доступ до персональних даних користувача.

Для вирішення цієї проблеми необхідно відредагувати файл .htacces, а саме додати наступний код:

```
<IfModule php5_module>
    php_flag session.cookie_httponly on
</IfModule>
```

Цей варіант підходить для серверів які використовують Apach. Apache – це програмне забезпечення, яке встановлено на сервер. Завдяки йому встановлюється з'єднання між користувачем, використовуючим браузер, і сервером, щоб здійснити передачу даних при запиті.

Також можливий ще один варіант вирішення, для цього необхідно додати атрибут “bool $httponly = TRUE” в функцію створення файлів Cookie.

3.3.2 Захист від Клікджейкінгу

Clickjacking – це атака, яка обманює користувачів, думаючи, що вони натискають одну річ, тоді як вони насправді натискають іншу. Його інша назва, переоформлення інтерфейсу користувача (UI), краще описує те, що відбувається. Користувачі думають, що вони використовують звичайний інтерфейс веб-сторінки, але насправді є прихований інтерфейс для керування; іншими словами, інтерфейс користувача було виправлено. Коли користувачі натискають щось, що вони вважають безпечним, прихований інтерфейс користувача виконує іншу дію.

Ідея цієї атаки дуже проста. Ось як clickjacking-атака була проведена на Facebook:

- Відвідувача заманюють на шкідливу сторінку (неважливо як);

- На сторінці є посилання, яке виглядає нешкідливо (наприклад, "Розбагатій прямо зараз" або "Натисни тут, це дуже смішно");
- Поверх цього посилання шкідлива сторінка розміщує прозорий <iframe> з src із сайту facebook.com таким чином, що кнопка «like» знаходиться прямо над цим посиланням. Зазвичай це робиться за допомогою z-index CSS;
- При спробі натиснути на це посилання відвідувач насправді натискає кнопку.

Для захисту від клікджейкінгу необхідно додати заголовок X-Frame-Options в файл .htacces на сервері, та вибрати одну з чотирьох опцію:

- DENY він блокує завантаження сайту у фрейм;
- SAMEORIGIN дозволяє завантажувати домен на рівні одного сайту;
- ALLOW-FROM your_domain дає можливість завантажувати домен в iframe певного домену;
- ALLOW-ALL значення за замовчуванням що дає можливість атакувати.

Було обрано опцію SAMEORIGIN, що дажє змогу завантажувати домен в фрейм того ж домену, та забороняє завантажувати інші, тому як даному проекні в цьому немає необхідності.

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Заходи щодо захисту установки від короткого замикання

Коротке замикання – це дотик, зіткнення струмоведучих різнополюсних жил (з різними потенціалами) або частин електропроводки, з виникненням спалаху від плазмової дуги і можливим розбризкуванням розплавленого металу провідників. Це короткий опис простими словами. Дуже часто коротке замикання є причиною пожежі. Скорочено таке замикання позначають абревіатурою КЗ.

Наприклад, в розетці довгий час щось іскрить і потріскує, гріється, а може і оплавляється вилка дроту електроприладу. Потім в розетці виникає хлопок, можливо, гасне світло – весь об’єкт знеструмлений у зв’язку із запобіжним спрацюванням автоматів в електрощиті. Причина всього цього – обвуглювання ізоляції проводів в розетці, торкання фазового проводу і нульового проводу. У місці дотику проводів виникає дуже маленький опір і вся велика сила струму, що знаходиться в ланцюгу, безперешкодно кинулася з усією висоти різниці потенціалів в нульовий провідник. У точці дотику провідників метал миттєво і сильно розігрівається, виникає плазмова дуга (як у електрода при роботі зварювальним апаратом), розплавлений метал розлітається в різні боки. Добре, якщо поруч немає легкозаймистих матеріалів або речовин. Наявність легкозаймистих речовин або матеріалів може призвести до виникнення пожежі.

Основними заходами захисту від ураження електричним струмом є:

- забезпечення недоступності струмопровідних частин, що перебувають під напругою, для випадкового дотику;
- електричний поділ мережі;
- усунення небезпеки ураження з появою напруги на корпусах, кожухах та інших частинах електроустаткування, що досягається захисним заземленням, зануленням, захисним відключенням;
- застосування малих напруг;

- захист від випадкового дотику до струмопровідних частин застосуванням кожухів, огорож, подвійної ізоляції;
- захист від небезпеки при переході з вищої на нижчу напругу;
- контроль і профілактика пошкоджень ізоляції;
- компенсація ємнісної складової струму замикання на землю;
- застосування спеціальних електрозахисних засобів - переносних приладів і запобіжних пристроїв;
- організація безпечної експлуатації електроустановок.

Електробезпека – це система організаційних і технічних заходів, що забезпечують захист людей від небезпечної і шкідливої дії електричного струму, електричної дуги, електромагнітного поля, статичної електрики.

Електричний розподіл мережі. Розгалужена мережа великої довжини має значну ємність і малий активний опір ізоляції щодо землі. Струм замикання на землю в такій мережі може бути значним. Якщо єдину сильно розгалужену мережу з великою ємністю і малим опором ізоляції розділити на ряд невеликих мереж такої самої напруги, які матимуть незначну ємність і високий опір ізоляції, небезпека ураження різко знизиться. Звичайно електричний розподіл мереж здійснюється шляхом підключення електроприймачів через розподільний трансформатор окремих електроприймачів, що живляться від основної розгалуженої мережі.

Захист від небезпеки при переході з вищої напруги на нижчу. При пошкодженні ізоляції між обмотками вищої і нижчої напруг трансформатора виникає небезпека переходу напруги і, як наслідок, небезпека ураження людини, виникнення займання і пожеж. Способи захисту залежать від режиму нейтралі. Мережі напругою до 1000 В з ізольованою нейтраллю, сполучені через трансформатор з мережами напругою вище за 1000 В, мають бути захищені пробивним запобіжником, установленим у нейтралі чи фазі з боку нижчої напруги трансформатора. Тоді у випадку пошкодження ізоляції між обмотками вищої і нижчої напруг цей запобіжник пробивається і нейтраль або фаза нижчої напруги заземлюється. Напруга нейтралі щодо землі $U_з = I_з * R_0$.

Заходом захисту є зниження цієї напруги до безпечного заземлення нейтралі з опором R0 <4 Ом.

Пробивні запобіжники застосовуються, коли вища напруга є більшою за 1000 В. Якщо вища напруга буде нижчою за 1000 В, пробивний запобіжник не спрацює. Тому вторинні обмотки знижувальних трансформаторів для живлення ручного електроінструмента і ручних ламп малою напругою заземлюють.

Контроль і профілактика пошкоджень ізоляції. Профілактика пошкоджень ізоляції спрямована на забезпечення її надійної роботи. Насамперед необхідно виключити механічні пошкодження, зволоження, хімічний вплив, запилення, перегріви. Але навіть у нормальних умовах ізоляція поступово втрачає свої початкові властивості, "старіє". З часом розвиваються місцеві дефекти. Опір ізоляції починає різко зменшуватися, а струм витоку - непропорційно зростати. У місці дефекту з'являються часткові розряди струму, ізоляція вигорає. Відбувається так званий пробій ізоляції, внаслідок чого виникає коротке замикання, що, у свою чергу, може спричинити пожежу чи ураження людей струмом.

Щоб підтримувати діелектричні властивості ізоляції, необхідно систематично виконувати профілактичні випробування, огляди, видаляти непридатну ізоляцію і заміняти її.

Періодично в приміщеннях без підвищеної небезпеки не рідше одного разу на два роки, а в небезпечних приміщеннях - кожні півроку перевіряють відповідність опору ізоляції нормі. При виявленні дефектів ізоляції, а також після монтажу мережі, її ремонту на окремих ділянках, відключення мережі між кожним проводом і землею та між проводами різних фаз проводять вимірювання.

Для вимірювання використовують прилад - мегаомметр на напруги 500, 1000, 2500 В з межами вимірів 0-100, 0-1000, 0-10000 МОм. Щоб мати уявлення ще й про опір ізоляції всієї мережі, вимірювання потрібно проводити під робочою напругою з підключеними споживачами. Такий контроль можливий тільки в мережах з ізольованою нейтраллю (у мережі з заземленою нейтраллю

постійний струм приладу контролю ізоляції замикається через заземлення нейтралі, і мегаомметр показуватиме нуль).

Застосовується також постійний (безперервний) контроль ізоляції - вимірювання опору ізоляції під робочою напругою протягом усього часу роботи електроустановки без автоматичного відключення. Відлік опору ізоляції здійснюється за шкалою приладу. При зниженні опору ізоляції до гранично допустимого чи нижче, прилад подає звуковий або світловий сигнал або обидва сигнали разом. З вітчизняних приладів контролю ізоляції найбільшого поширення одержали ПКІ, РУВ, УАКІ, М-143, МКН-380, Ф-419. Найпростішим засобом контролю ізоляції є вольтметр. В установках напругою до 1000 В вольтметри підключають безпосередньо до фаз, а в установках з напругою понад 1000 В - через вимірювальний трансформатор.

4.2 Вимоги ергономіки до організації робочого місця оператора ПК

Загальні ергономічні вимоги для організації робочого місця користувача ПЕОМ (ГОСТ 12.2.049-80, ГОСТ 122032-78, ГОСТ 22269-76). Ці вимоги встановлюють основні параметри робочого місця, оснащеного дисплеєм, і враховують особливість виконуваних робіт.

Параметри робочого місця повинні бути наступними: площа кабінету, в якому буде проходити робота повинна бути не менш 6 м², а об’єм не менш 24 м³. Для внутрішньої обробки приміщення повинні використовуватися дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі – 0,7-0,8; для стін – 0,5-0,6; для підлоги – 0,3-0,5.

Конструкція робочого столу повинна забезпечувати оптимальне розміщення на робочій поверхні використовуваного обладнання. Конструкція крісла повинна забезпечувати підтримку раціональної робочої пози під час роботи з відео-дисплейним терміналом (Далі ВДТ) і ПЕОМ, дозволяти змінювати позу з метою зниження статичного напруження м'язів шийно-плечової області і спини для попередження розвитку втоми працюючого (згідно з ГОСТ 12.2.032-78). Поверхня сидіння, спинки та інших елементів стільця

(крісла) повинна бути напівм'якою, з покриттям, що не електризується, неслизьке та повітронепроникне, що забезпечує легке очищення від забруднення.

Висота робочої поверхні столу, за відсутності можливості її регулювання повинна складати 725 мм. Робочий стіл повинен мати простір для ніг висотою не менше 600 мм, шириною – не менше 500 мм, не менше 450 мм в глибину на рівні колін і на рівні простягнутої ноги – не менше 650 мм. Робоче місце має бути обладнане підставкою для ніг, має ширину не менше 300 мм, глибину не менше 400 мм, регулювання по висоті в межах 150 мм за кутом нахилу опорної поверхні підставки до 20 градусів.

Відстань від очей користувача до екрану дисплея має становити 500-700 мм. Кут зору 10-20°, але не більше 40°; кут між верхнім краєм дисплея і рівнем очей користувача має становити не менше 10°. Кращим є розташування екрану перпендикулярно до лінії зору користувача.

Робочі місця по відношенню до світлових прорізів повинні розташовуватися не ближче 3 м так, щоб природне світло падало збоку, переважно зліва. Освітленість також впливає на стан здоров'я і працездатність людини. У відповідності зі СНіП 11-4-79 встановлені наступні вимоги до освітленості:

Для штучного освітлення:

- Комбіноване освітлення – освітленість 1500 лк;
- Загальне освітлення – освітленість 400 лк.

Для природного освітлення:

- Верхнє або комбіноване освітлення – коефіцієнт природної освітленості (далі КПО) 10%;
- Бічне освітлення – КПО 3.5%.

Для суміщеного освітлення:

- Верхнє або комбіноване освітлення – КПО 3-6%;
- Бічне освітлення – КПО 1.1-2%.

До основних показників, що визначають умови здорової роботи, належать: фон, контраст об'єкта з фоном, видимість, показник осліпленості, коефіцієнт пульсації освітленості.

Фон характеризується коефіцієнтом відбиття. Контраст об'єкта з фоном (К) характеризується співвідношенням яскравості розглянутого об'єкта (точки, лінії, знаки) і фону. Оскільки роботи користувача ПЕОМ відносяться до категорії 1а – легкі фізичні роботи (роботи проводяться сидячи і супроводжуються незначним фізичним напруженням, з енерговитратами до 120 ккал / годину), необхідно дотримуватися наступних норм: коефіцієнт відображення більше 0,4, тобто світлий фон; контраст об'єкта з фоном великий і середній при К більше 0,2 (згідно СНіП 11-4-79).

У полі зору користувача ПЕОМ має бути забезпечений відповідний розподіл яскравості. Відношення яскравості екрана до яскравості оточуючих його поверхонь не повинно перевищувати у робочій зоні 3:1 (СНіП 11-4-79). У зв'язку з цим дисплей ПЕОМ повинен відповідати наступним вимогам:

- Яскравість свічення екрану не менше 100 кд/м;
- Мінімальний розмір світної точки для кольорового дисплея не більше 0,6 мм;
- Контрастність зображення знаку – не менше 0,8;
- Низькочастотне тремтіння зображення в діапазоні 0,05-1,0 Гц повинно знаходитися в межах 0,1 мм;
- Екран повинен мати покриття антивідблиску;
- Відеомонітор повинен бути обладнаний поворотним майданчиком, що дозволяє переміщати відеотермінал в горизонтальній і вертикальній площинах в межах 130-220 мм і змінювати кут нахилу на 10-15 мм.

Коефіцієнт відбиття світла матеріалами і обладнанням всередині приміщень має велике значення для освітлення: чим більше світла відбивається від поверхонь, тим вище освітленість. Коефіцієнт відображення відповідно повинен бути для: стелі 60-70%, стін 40-50%, підлоги 30%, для інших поверхонь 30-40%.

Результати досліджень показують, що найбільшою мірою негативний фізіологічний вплив на операторів ПК пов'язаний з дискомфортними зоровими умовами через неправильно спроектоване освітлення. Згідно СНіП II-4-79 освітленість на горизонтальній площині робочого місця оператора ЕОМ повинна складати 400 лк при висоті цій площині 0,8 м над підлогою.

Світловий клімат визначає зоровий дискомфорт. Запобігти шкідливому впливу освітлення можна шляхом правильного підбору системи освітлення, джерел світла (за їх спектрального складу випромінювання), світильників. Коли штучне світло змішується з природним, рекомендується використовувати лампи за спектральним складом найбільш близькі до сонячного світла. Світильники слід вибирати з розсіювачами, а блискучі деталі освітлювального обладнання, що можуть потрапити в поле зору оператора, повинні бути замінені на матові.

Розташовувати робоче місце, обладнане дисплеєм, необхідно таким чином, щоб у полі зору оператора не потрапляли вікна або освітлювальні прилади; вони не повинні знаходиться і безпосередньо за спиною оператора. Вікна в приміщеннях з дисплеями обладнують шторами з коефіцієнтом відображення 0,5 ... 0,7, стіни фарбують матовою фарбою з коефіцієнтом відображення 0,4 ... 0,6.

Світловий клімат може бути поліпшений шляхом встановлення спеціальних антивідблискових контрастних фільтрів, однак при виборі типу фільтра необхідно враховувати умови роботи з комп'ютером, оскільки оптимальні значення коефіцієнтів пропускання і дзеркального відображення фільтрів залежать від освітленості робочого місця і типу джерела світла.

Враховуючи великий вплив освітлення на працездатність оператора при роботі з комп'ютером, проведемо розрахунок необхідної освітленості в приміщенні з дисплеями при наступних умовах: гігієнічна норма освітленості на горизонтальній поверхні на рівні робочого місця оператора – 400 лк; ширина приміщення – 7 м, довжина – 8 м, висота – 3 м. Коефіцієнт відбиття від стелі –

70, від стін – 50, від робочих поверхонь – 30. Повітряне середовище – нормальне (вміст пилу, диму й кіптяви не більше 5 мг/м³).

Повітряне середовище в робочій зоні визначається мікрокліматом виробничого приміщення. Величини температури, відносної вологості та швидкості руху повітря на робочих місцях з дисплеями повинні відповідати допустимим значенням, які встановлені ГОСТ 12.1.005-88 ССБТ для категорії робіт 1а (легкі фізичні роботи, вироблені сидячи і супроводжуються незначною фізичною напругою до 120 ккал/год.). Згідно з цим документом допустимі значення температури повітря в приміщенні становлять 19-25 °С, відносної вологості повітря – 55%, швидкості руху повітря на рівні особи – 0,1 м/С. При наявності досить комфортного робочого середовища атмосферний тиск по ГОСТ 21552-84 ССБТ може змінюватися від 84 до 107 кПа (630 ... 800 мм рт. ст.).

Шум несприятливий для людини, особливо при тривалому впливі. В оператора це виражається в зниженні працездатності (наприклад, швидкість обробки тексту зменшується на 10-15%), у прискоренні розвитку зорового стомлення, зміну відчуття кольору, підвищенні витрати енергії (на 17%).

Тривалий та інтенсивний шум значно знижує продуктивність праці і призводить до зростання кількості помилок у роботі. У відділі головного економіста шум може створюватися телефонними дзвінками та розмовами, системними блоками та клавіатурою ПЕОМ. Так само джерелами шуму можуть бути системи кондиціювання та вентилювання повітря, існують і зовнішні джерела шуму (наприклад, працюють агрегати на вулиці).

ВИСНОВКИ

При написанні даного дипломного проекту було вивчено багато нового про структури види типи та нюанси написання оформлення та написання прогрманого коду сайта, його захисту, для роботи з базою та інших функцій. Даний дипломний проект дав змогу навчитися правильно редагувати сайт за допомогою CSS, сформувати його структуру за допомогою HTML та створювати функціонал на стороні клієнта за допомогою скриптової мови програмування JavaScript, реалізувати захист від можливих загроз. За допомогою JavaScript на сайті реалізована корзана, слайдер, та модальні вікна.

Розроблено панель адміністратора, яка написана на мові програмування PHP. В ній реалізовано додавання, редагування та видалення товарів з сайту, а також видалення відгуків написаних користувачами. На сайті реалізована можливість реєстрації та авторизації для користувачів, та можливість залишати відгуки.

Сайт захищено від DDoS-атак за допомогою сервісу Cloudflare, персональні дані користувачів захищені SSL сертифікатом, налаштовано щомісячне резервне копіювання баз даних, проведено сканування можливих загроз за допомогою сканра Nikto, проаналізовано загрози, реалізовано захист від виявлених загроз.

Користувачі також мають можливість оплачувати товари за допомогою банківських карт або за допомогою мобільного додатку Privat24. Ця функція реалізована за допомогою безкоштовного LIQPAY API.

Також проведено тестування веб-сайту згідно тест-плану, були протестовані усі функції сайту, виправлено усі синтаксичні помилки HTMT та CSS, всі сторінки сайту є зрозумілими і простими у використанні, доступ до головного меню здійснюється зі всіх сторінок сайту. Сайт протестовано у всіх сучасних веб-переглядачах та на всіх популярних розширеннях екрану.

Оскільки програмний продукт згідно даного дипломного проекту був успішно виконаний та вдало функціонує – можна зробити висновок, що всі перераховані завдання виконано.

ПЕРЕЛІК ПОСИЛАНЬ

1. Hostinger | [Електронний ресурс] – режим доступу до ресурсу: https://hostinger.com.ua/ Дата доступу: 21.05.22

2. LiqPay – Миттєві платежі | [Електронний ресурс] – режим доступу до ресурсу: https://www.liqpay.ua/ru Дата доступу: 21.05.22

3. PHP – найбільш популярна мова для веб програмування | Chili-web | [Електронний ресурс] – режим доступу до ресурсу: https://chili-web.com.ua/php-5/ Дата доступу: 22.05.22

4. Документація програмного забезпечення – Вікіпедія | [Електронний ресурс] – режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D1%96%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F Дата доступу: 23.05.22

5. Етапи створення веб-сайтів. Основи веб-дизайну - Студопедия.Орг | [Електронний ресурс] – режим доступу до ресурсу: https://studopedia.org/9-62702.html Дата доступу: 24.05.22

6. Історія створення першого в світі сайту - Web Building | [Електронний ресурс] – режим доступу до ресурсу: https://webbuilding.com.ua/ukr/articles/istoriya-stvorennya-pershogo-saytu/ Дата доступу: 21.05.22

7. Какие бывают интернет-магазины: классификация – ImageCMS | [Електронний ресурс] – режим доступу до ресурсу: https://www.imagecms.net/blog/e-commerce/kakie-byvaiut-internet-magaziny-klassifikatsiia Дата доступу: 25.05.22

8. Переваги інтернет-магазинів | Компанія ВебМарк | [Електронний ресурс] – режим доступу до ресурсу: http://webmark.com.ua/ua/ecommerce/web-store-benefits.html Дата доступу: 26.05.22

9. Робота за комп'ютером: пільги і компенсації, передбачені ... | [Електронний ресурс] – режим доступу до ресурсу: http://cons.parus.ua/_d.asp?r=06FT523ebaa42fdbeece00111fc9e6368165b Дата доступу: 26.05.22

10. Створення та Наповнення сайтів контентом ⊳ WEBMAESTRO | [Електронний ресурс] – режим доступу до ресурсу: https://webmaestro.com.ua/ua/blog/napovnennia-saitu/ Дата доступу: 26.05.22

11. Типи сайтів. Частина 3. Інтернет-магазин | Web-LightHouse | [Електронний ресурс] – режим доступу до ресурсу: https://web-lighthouse.com/news/типи-сайтів-частина-3-інтернет-магазин/ Дата доступу: 27.05.22

12. Що таке SSL сертифікат і навщо він потрібен - HOSTiQ | [Електронний ресурс] – режим доступу до ресурсу: https://hostiq.ua/ukr/info/what-is-ssl/ Дата доступу: 26.05.22

13. Як захтстити сайт від DDoS-атак? - Datami | [Електронний ресурс] – режим доступу до ресурсу: https://datami.ua/ru/kak-zashhitit-sajt-ot-ddos-atak/ Дата доступу: 27.05.22