

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

Факультет комп'ютерно-інформаційних систем і програмної інженерії  
(повна назва факультету)

Кафедра комп'ютерних наук  
(повна назва кафедри)

# КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка веб-сайту магазину інструментів "PROF" з використанням  
технологій JavaScript

Виконав: студент IV курсу, групи СН-42

спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

(підпис)

Станкевич М.В.

(прізвище та ініціали)

Керівник

(підпис)

Литвиненко Я.В.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Шимчук Г.В.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Боднарчук І.О.

(прізвище та ініціали)

Рецензент

(підпис)

Золотий Р.З.

(прізвище та ініціали)

Тернопіль  
2024

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

Факультет комп'ютерно-інформаційних систем і програмної інженерії  
(повна назва факультету)

Кафедра комп'ютерних наук  
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.  
(підпис) (прізвище та ініціали)

« 26 » червня 2024 р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр  
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки  
(шифр і назва спеціальності)

Студенту Станкевич Михайло Володимирович  
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка веб-сайту магазину інструментів "PROF" з використанням технологій JavaScript

Керівник роботи Литвиненко Ярослав Володимирович, д.т.н., про. професор кафедри КН  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 29 » квітня 2024 року № 4/7-470

2. Термін подання студентом завершеної роботи 26.06.2024р.

3. Вихідні дані до роботи Структура сайту, перелік необхідної інформації яка буде розміщена на сайті

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Розгляд типових рішень та засобів для розробки веб-сайту. 1.1 Огляд типових технологій які будемо використовувати при розробці веб-сайту. 1.2 Редактори для розробки веб-сату. 1.3 Система управління базою даних. 1.4 Мова гіпертекстової розмітки HTML. 1.5 Каскадні таблиці стилів CSS. Висновок до першого розділу.

РОЗДІЛ 2. Розробка веб-сайту. 2.1 Опис загальних кроків розробки веб-сайту. 2.2 Дизайну веб-сайту та його розробка. 2.3 Тестування веб-сайту. 2.4 Висновок до другого розділу.

РОЗДІЛ 3. Безпека життєдіяльності, основи охорони праці. 3.1 Безпека життєдіяльності (БЖД) наука, що вивчає проблеми безпечного перебування людини в довкіллі в процесі різних видів її діяльності. 3.2 Основні небезпеки в урбанізованому середовищі. 3.3 Загальні закономірності виникнення небезпек. 3.4 Висновок до третього розділу. Висновки. Перелік джерел

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

Слайди: Тема 1. Мета та задачі 2. Структурна схема веб-сайту 3. Вигляд головної сторінки 4. вигляд другої сторінки 5. Вигляд третьої сторінки 6. Вигляд сторінки «Контакти» 7. Висновки 8.



## АНОТАЦІЯ

Розробка веб-сайту магазину інструментів "PROF" з використанням технологій JavaScript // Кваліфікаційна робота освітнього рівня «Бакалавр» // Станкевич Михайло Володимирович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СН-42 // Тернопіль, 2024 // С. , рис. – , табл. – , кресл. – , додат. – , бібліогр. – .

**Ключові слова:** web-сайт, сайт, JavaScript скрипт, браузер, web-сторінка, сервер.

Кваліфікаційна робота присвячена розробці веб-сайту інтернет-магазину інструментів "PROF" з використанням технологій JavaScript. В роботі описані поняття, методи принципи необхідні для створення web-сайтів, наведено використання необхідних програм для створення веб-сайту і бази даних MongoDB.

В першому розділі кваліфікаційної роботи описано огляд технологій, які будуть використовуватись для розробки веб-сайту. Зокрема описано технологію JavaScript. Описані відомі фреймворки здійснено порівняння технологій MySQL та MongoDB. У другому розділі наведено опис та розробку безпосередньо веб-сайту, зокрема структуру та описані основні сторінки веб-сайту та його елементи.

У третьому розділі кваліфікаційної роботи описано основні питання безпеки життєдіяльності, а саме БЖД як наука, що вивчає проблеми безпечного перебування людини в довкіллі в процесі різних видів її діяльності. Крім цього наведені відомості про основні небезпеки в урбанізованому середовищі, а також рятувальні закономірності виникнення небезпек.

Об'єкт дослідження: процес розробки веб-сайту.

Предмет дослідження: методи та технології які застосовуються при розробці веб-сайтів.

Практичне значення одержаних результатів полягає у створенні веб-сайту для інтернет-магазину інструментів "PROF".

## ANNOTATION

Development of the "PROF" Tool Store Website Using JavaScript Technologies // Qualification work of the educational level "Bachelor" // Stankevich Mykhailo Volodymyrovych // Ivan Pulyuy Ternopil National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer . of computer sciences, group SN-42 // Ternopil, 2024 // C. , fig. - , tab. - , armchair. - , add. – , bibliography -

**Keywords:** website, site, JavaScript script, browser, web page, server.

The qualification work is dedicated to the development of the website of the online tools store "PROF" using JavaScript technologies. The work describes the concepts, the necessary principles of creating websites, the use of one program for creating a website and a MySQL database is given.

The first section of the qualification work describes the overview of technologies. which will be used to develop the website. which describes JavaScript technology. Known frameworks of MySQL technology are described. The second section provides a description and design of each website, including the structure and main pages of the website and its elements.

The third section of the qualification work describes the main issues of life safety, namely BZD as a science that studies the problems of a person's safe stay in the environment in various types of his activities. In addition to these given information about the main dangers in the urban environment, as well as the rescue patterns of the occurrence of dangers.

Research object: website development process.

Research subject: technologies and methods used in website development.

The practical significance of the obtained results arose in the modern website for the online tool store "PROF".

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ**

БЖД – Безпека життєдіяльності.

СУБД – Система управління базами даних.

CSS – Cascading Style Sheets

HTML – HyperText Markup Language.

## ЗМІСТ

ВСТУП .....	8
РОЗДІЛ 1. РОЗГЛЯД ТИПОВИХ РІШЕНЬ ТА ЗАСОБІВ ДЛЯ РОЗРОБКИ ВЕБ-САЙТУ .....	9
1.1 Огляд типових технологій для розробки веб-сайту .....	9
1.2 Редактори для написання JavaScript при розробці веб-сайту .....	10
1.3 Огляд систем управління базою даних .....	11
1.4 Мова гіпертекстової розмітки HTML .....	13
1.5 Каскадні таблиці стилів CSS .....	14
1.6 JavaScript мова програмування для веб-розробки .....	16
1.7 Фреймворки та бібліотеки для створення веб-додатків з JavaScript..	17
1.8 Фреймворк React.js .....	20
1.9 Фреймворк Node.js .....	22
1.10 Система керування базою даних MongoDB .....	23
1.11 Порівняння MongoDB і MySQL .....	24
1.12 Висновок до першого розділу .....	26
РОЗДІЛ 2. РОЗРОБКА ВЕБ-САЙТУ .....	27
2.1 Опис загальних кроків розробки веб-сайту .....	27
2.2 Дизайну веб-сайту та його розробка .....	28
2.3 Тестування веб-сайту .....	34
2.4 Висновок до другого розділу .....	36
РОЗДІЛ 3. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	38
3.1 Безпека життєдіяльності (БЖД) наука, що вивчає проблеми безпечного перебування людини в довкіллі в процесі різних видів її діяльності.....	38
3.2 Основні небезпеки в урбанізованому середовищі .....	40
3.3 Загальні закономірності виникнення небезпек.....	42
3.4 Висновок до третього розділу .....	46
ВИСНОВКИ.....	47



ПЕРЕЛІК ДЖЕРЕЛ.....	48
ДОДАТКИ	

## ВСТУП

**Актуальність теми.** Кожен сучасний бізнес повинен мати можливість торгувати не лише в межах одного міста. З цим може допомогти використання веб-сайтів. Наявність інтернет-магазину вирішує багато завдань наприклад окрім того що дозволяє торгувати ще дозволяє взаємодіяти з покупцями чи оптовими клієнтами. Тому створення веб-сайту для інтернет-магазину є важливим та актуальним кроком.

**Мета і задачі дослідження.** Метою даної кваліфікаційної роботи освітнього рівня «Бакалавр» є розробка веб-сайту інтернет-магазину інструментів "PROF" з використанням технологій JavaScript. Для досягнення поставленої мети потрібно виконати ряд завдань, зокрема:

- Провести аналіз подібних сайтів;
- Проаналізувати технології які будуть використовуватись для розробки веб-сайту та вибрати конкретні;
- Здійснити розробку дизайну веб-сайту;
- Здійснити розробку структури сторінок веб-сайту;
- Здійснити розробку необхідних елементів керування веб-сайтом;
- Провести тестування розробленого веб-сайту.

### **Практичне значення одержаних результатів.**

Інтернет-сайт для магазину інструментів "PROF" є важливим оскільки дозволяє здійснити продажі продукції за межами регіону де він знаходиться.

Наявність веб-сайту магазину інструментів "PROF" дозволяє надати більше різної інформації про продукцію магазину її особливості та ціну, тут можна отримати консультацію фахівця з продажу інструментів, зробити замовлення дізнатись про доставку. Тому створення веб-сайту має практичне значення для магазину і є актуальним та важливим.

## РОЗДІЛ 1. РОЗГЛЯД ТИПОВИХ РІШЕНЬ ТА ЗАСОБІВ ДЛЯ РОЗРОБКИ ВЕБ-САЙТУ

### 1.1 Огляд типових технологій для розробки веб-сайту

Для розробки веб-сайту з використанням JavaScript існує багато технологій і інструментів, які можна використовувати залежно від потреб проекту. Розглянемо основні технології і інструменти, які можна використати:

Фреймворки для клієнтської частини:

**React.js:** Це потужний фреймворк для створення інтерфейсів користувача. React дозволяє створювати компоненти, які можна повторно використовувати, і використовує віртуальний DOM для ефективного оновлення інтерфейсу.

**Vue.js:** Легкий і простий у використанні фреймворк, який також дозволяє створювати компоненти і має добру підтримку для реактивного програмування.

**Angular:** Інша популярна платформа для створення веб-додатків, що надає багато можливостей, включаючи вбудований механізм відлагодження, директиви і сервіси.

Бібліотеки і утиліти:

**jQuery:** Легковагова JavaScript бібліотека для спрощення маніпуляцій з DOM, анімацій, обробки подій та AJAX запитів.

**Bootstrap:** HTML, CSS і JavaScript фреймворк для швидкої розробки адаптивних і стильних веб-інтерфейсів.

**Tailwind CSS:** Сучасний CSS фреймворк, який дозволяє створювати кастомізовані дизайни без необхідності написання CSS.

Інструменти для розробки і збудови:

**Webpack:** Модульний збирач для JavaScript і його додатків. Він дозволяє збирати різні модулі в один або декілька пакетів, щоб зменшити завантаження сторінки.

Babel: Транслятор JavaScript, який дозволяє використовувати сучасні функції JavaScript (наприклад, ES6 і вище) у всіх браузерах.

Системи керування версіями і інші інструменти:

Git: Система керування версіями для контролю версій вашого коду і співпраці з іншими розробниками.

npm (Node Package Manager): Менеджер пакетів для JavaScript, який дозволяє встановлювати, оновлювати та використовувати різноманітні пакети і бібліотеки для вашого проекту.

Ці технології і інструменти є основними для сучасної веб-розробки з використанням JavaScript. Розглянемо вибір конкретних технологій які будемо використовувати в роботі.

## **1.2 Редактори для написання JavaScript при розробці веб-сату**

Для редагування JavaScript коду на веб-сайтах можна використовувати різні редактори, які надають зручне середовище розробки з підсвічуванням синтаксису, автодоповненням, підтримкою вбудованих інструментів і багато іншого. Розглянемо поширені редактори, які часто використовуються для розробки JavaScript веб-сайтів:

Visual Studio Code (VS Code):

Особливості: VS Code є потужним і легким використанні редактором, який підтримує розширення для майже будь-якої мови програмування, включаючи JavaScript. Він має вбудовану підтримку Git, підсвічування синтаксису, автодоповнення, вбудовані термінали і інше.

Sublime Text:

Особливості: Sublime Text — це швидкий, легкий і дуже налаштовуваний текстовий редактор, який підтримує багато розширень для покращення функціональності. Він також має підсвічування синтаксису, автодоповнення та можливості розширювати його функціонал.

Atom:

Особливості: Atom є відкритим і розширюваним редактором, створеним GitHub. Він підтримує підсвічування синтаксису, автодоповнення, вбудовані Git і GitHub підтримку, а також велику кількість розширень для веб-розробки.

WebStorm:

Особливості: WebStorm — це інтегроване середовище розробки (IDE) від JetBrains, спеціалізоване на розробці веб-додатків, включаючи JavaScript, HTML і CSS. Воно має багато інструментів для автоматизації рутинних завдань, рефакторингу коду, підсвічування синтаксису і багато іншого.

Brackets:

Особливості: Brackets — це безкоштовний текстовий редактор від Adobe, спеціально розроблений для веб-розробки. Він має вбудовані інструменти для прямого редагування CSS, вбудовані прев'ю для HTML і CSS, підсвічування синтаксису для JavaScript та інші корисні функції.

Ці редактори є дуже популярними серед веб-розробників і надають різноманітні інструменти для зручного редагування JavaScript коду.

### **1.3 Огляд систем управління базою даних**

Для веб-сайтів існують різноманітні системи керування базами даних (СКБД), які використовуються для зберігання, керування і доступу до даних. Розглянемо поширені СКБД для веб-розробки:

MySQL:

Тип: Реляційна база даних (RDBMS).

Особливості: MySQL є однією з найбільш поширених відкритих реляційних баз даних. Вона підтримує SQL для маніпуляції даними, має високу швидкодію, надійність і добре підходить для традиційних веб-додатків, таких як блоги, електронна комерція, управління вмістом і т. д.

PostgreSQL:

Тип: Реляційна база даних (RDBMS).

Особливості: PostgreSQL також є відкритою реляційною базою даних і володіє дуже потужними можливостями, включаючи підтримку складних запитів, транзакцій, реплікації і розширення функціональності через розширення.

MongoDB:

Тип: Документ-орієнтована база даних (NoSQL).

Особливості: MongoDB використовує JSON-подібні документи для зберігання даних у вигляді BSON (Binary JSON). Вона надає гнучкість у роботі з даними без суворої схеми і добре підходить для сучасних веб-додатків, які вимагають швидкої розробки і гнучкості.

SQLite:

Тип: Вбудована реляційна база даних (RDBMS).

Особливості: SQLite є легкою, вбудованою СКБД, яка не вимагає окремого сервера. Вона ідеально підходить для невеликих проектів або мобільних додатків, де не потрібна велика масштабованість.

Microsoft SQL Server:

Тип: Реляційна база даних (RDBMS).

Особливості: SQL Server є платною реляційною базою даних від Microsoft, яка має широкі можливості для розробки веб-додатків на платформі Windows. Вона підтримує високу доступність, реплікацію і інтеграцію з іншими продуктами Microsoft.

Ці системи керування базами даних мають свої особливості і варіанти застосування в залежності від потреб конкретного проекту. Вибір конкретної СКБД залежить від таких чинників, як тип даних, масштаб проекту, вимоги до швидкодії, доступність ресурсів і власних уподобань. Враховуючи це виберемо MongoDB оскільки вона надає гнучкість у роботі з даними без закріпленої структури і підходить для типових веб-сайтів.

## 1.4 Мова гіпертекстової розмітки HTML

Для написання веб сайту необхідно згадати за мову гупертекстової розмітки. Тому, HTML5, як остання версія мови розмітки HTML, внесла багато нововведень і покращень у порівнянні з попередніми версіями. Важливі особливості мови розмітки HTML5:

Нові семантичні елементи:

HTML5 вводить багато нових семантичних елементів, які дозволяють краще структурувати веб-сторінки без необхідності використання загальних блоків. Наприклад: `<header>`, `<footer>`, `<nav>`, `<section>`, `<article>`, `<aside>`, `<main>` і т.д.

Підтримка аудіо і відео:

HTML5 включає в себе нові теги `<audio>` і `<video>`, які дозволяють вбудовувати аудіо і відео контент без необхідності використання сторонніх плагінів, таких як Flash. Ці теги підтримують різні формати медіа і мають вбудовані можливості для управління відтворенням і відображенням контрольних елементів.

Покращена підтримка форм:

HTML5 включає нові атрибути і типи для елемента `<input>`, що полегшує створення форм. Наприклад, атрибути `placeholder`, `required`, `pattern` дозволяють здійснювати валідацію даних без використання JavaScript. Також додано нові типи введення, такі як `email`, `url`, `tel`, `date`, `time` і т.д.

Покращена підтримка для веб-додатків:

HTML5 пропонує нові можливості для розробки веб-додатків, такі як локальне сховище (`localStorage` і `sessionStorage`), які дозволяють зберігати дані локально в браузері. Також є підтримка для роботи в офлайн-режимі за допомогою кешування (`Application Cache API`).

Графіка і векторні зображення:

HTML5 включає підтримку нових елементів `<canvas>` і `<svg>`. `<canvas>` дозволяє програмно малювати графіку, що відображається безпосередньо в

браузері, в той час як `<svg>` використовує векторні графічні об'єкти, які масштабуються без втрати якості.

Підтримка нових атрибутів і характеристик:

HTML5 включає ряд нових атрибутів для різних тегів, таких як `autocomplete`, `autofocus`, `contenteditable`, `download`, `hidden`, `spellcheck` і т.д. Ці атрибути дозволяють забезпечити кращий контроль і функціональність сторінки без необхідності використання JavaScript.

Підтримка WebGL:

HTML5 підтримує WebGL, що є JavaScript API для рендерингу 3D і 2D графіки в браузері без використання плагінів.

Узагальнення і підтримка мобільних пристроїв:

HTML5 враховує особливості роботи з мобільними пристроями, включаючи підтримку тач-івентів, медіа-запити для адаптивного дизайну і інші технології, що полегшують розробку мобільних додатків і адаптивних веб-сторінок.

HTML5 відкриває багато нових можливостей для розробників веб-додатків і спрощує створення сучасних, багатофункціональних і семантично багатих веб-сторінок.

## **1.5 Каскадні таблиці стилів CSS.**

Поряд мовою гупертекстової розмітки використовують каскадні таблиці стилів. Тому, опишемо CSS 3 — це третя версія Cascading Style Sheets, яка включає в себе багато нововведень і покращень у порівнянні з попередніми версіями. Ось деякі ключові особливості CSS3:

Модульність:

CSS3 розділений на різні модулі, що дозволяє використовувати тільки потрібні функції без завантаження всього стандарту. Це полегшує розширення мови і реалізацію нових можливостей поступово.

Нові властивості для розмітки:



**Flexbox:** Модель розмітки, яка дозволяє елементам розміщатися в контейнері з урахуванням властивостей їхнього розміщення, вирівнювання і розподілу внутрішнього простору.

**Grid:** Сіткова модель, яка надає більш гнучкий і складний підхід до розміщення елементів у веб-сторінці з допомогою сітки.

**Анімації і переходи:**

**Анімації (CSS Animations):** Дозволяють створювати плавні і анімовані переходи між станами елементів без використання JavaScript. Використовуються ключові фрейми (keyframes) для визначення анімації.

**Переходи (CSS Transitions):** Дозволяють контролювати зміну стилів елементів при зміні їхнього стану (наприклад, при наведенні курсора миші).

**Тіні і округлення кутів:**

Додані нові властивості для керування тінями (box-shadow) і округленням кутів (border-radius), що робить стилізацію елементів більш гнучкою і естетичною.

**Граденти і фонові зображення:**

**Граденти (CSS Gradients):** Дозволяють створювати плавні переходи між кольорами на елементах, що замінюють зображення градієнтами.

**Фонові зображення (CSS Backgrounds):** Нові властивості для керування фоновими зображеннями, включаючи можливість розміщення, повторення, масштабування та інші ефекти.

**Текстові ефекти:**

**Тінь тексту (text-shadow):** Дозволяє додавати тіні до тексту для покращення його читабельності або естетичного ефекту.

**Обтікання тексту (text-wrap):** Нові властивості для керування тим, як текст обтікає зображення або інші елементи.

**Гнучкість і адаптивний дизайн:**

CSS3 надає більше можливостей для адаптивного дизайну, включаючи медіа-запити (media queries), які дозволяють змінювати стиль веб-сторінки в залежності від характеристик пристрою, таких як розмір екрану.

Нові типи шрифтів:

Додані нові властивості для керування шрифтами, такі як `@font-face`, що дозволяють використовувати власні шрифти на веб-сторінках, а також `font-feature-settings` для активування або вимкнення окремих шрифтових функцій.

Ці особливості CSS3 роблять веб-розробку більш простою, ефективною і естетичною, дозволяючи створювати складні інтерфейси і забезпечувати покращену користувацьку взаємодію без великої залежності від JavaScript.

## **1.6 JavaScript мова програмування для веб-розробки**

Перейдемо до опису мови JavaScript, вона є ключовою мовою програмування для веб-розробки і має багато особливостей, які роблять її незамінною для створення динамічних інтерактивних веб-сайтів. Ось деякі з основних особливостей JavaScript для веб-сайтів:

Клієнтська мова програмування:

JavaScript виконується безпосередньо в браузері користувача, що дозволяє змінювати контент і взаємодіяти з користувачем без необхідності звертатися до сервера. Це робить його основним інструментом для створення динамічних інтерфейсів.

Обробка подій:

JavaScript дозволяє визначати реакції на події, такі як клік мишею, натискання клавіші, завантаження сторінки і т.д. Це робить можливим створення інтерактивних веб-сайтів, де дії користувача впливають на вміст сторінки.

Маніпуляція DOM:

JavaScript дає можливість змінювати структуру і стилі HTML і CSS на сторінці, що відомо як маніпуляція Document Object Model (DOM). Це дозволяє динамічно оновлювати вміст сторінки без перезавантаження.

Асинхронне програмування:

JavaScript підтримує асинхронне програмування за допомогою колбеків, промісів і `async/await`, що дозволяє виконувати операції, які не блокують інші процеси. Це особливо важливо для роботи з введенням/виведенням (I/O), таким як запити до сервера AJAX.

Розширення можливостей браузера:

JavaScript використовується для розширення можливостей браузера через використання веб-API. Наприклад, API для роботи з геолокацією, доступом до камери та мікрофону, роботи з файлами, рендерингу 2D/3D графіки за допомогою WebGL і багато інших.

Фреймворки і бібліотеки:

JavaScript має велику екосистему фреймворків і бібліотек, які полегшують розробку веб-додатків. Наприклад, Angular, React, Vue.js для розробки клієнтських частин, а також Node.js для серверної розробки на JavaScript.

Кросплатформеність:

JavaScript підтримується практично всіма сучасними браузерами і може виконуватися на різних платформах, включаючи Windows, macOS і Linux. Це робить його універсальним інструментом для веб-розробки.

JavaScript є мовою з великим потенціалом, яка постійно розвивається і адаптується до нових вимог веб-розробки, що робить її незамінною для створення сучасних веб-додатків і сайтів.

## **1.7 Фреймворки та бібліотеки для створення веб-додатків з JavaScript**

Розглянемо фреймворки та бібліотеки. Фреймворки дозволяють вам швидше створювати веб-додатки з JavaScript. Наведемо лише найпопулярніші фреймворки і бібліотеки:

**React:** Для створення інтерфейсів користувача. Рекомендується використовувати разом з бібліотекою для стану, наприклад, Redux або MobX.

Angular: Інтегрований фреймворк, який надає рішення для всієї стеку розробки.

Vue.js: Прогресивний JavaScript фреймворк для створення інтерфейсів користувача.

jQuery: Легковагова бібліотека, яка спрощує маніпулювання DOM і взаємодію з AJAX.

Express.js: Для створення серверних додатків на Node.js.

Управління пакетами: Для управління залежностями JavaScript проекту використовуються менеджери пакетів. Найпопулярніші — npm (Node Package Manager) і Yarn.

Системи контролю версій: Для керування версіями коду і спільної роботи над проектом використовуються системи контролю версій, такі як Git (і платформи для роботи з ним, наприклад, GitHub, GitLab, Bitbucket).

Інструменти тестування: Для тестування коду на JavaScript використовуються різноманітні фреймворки і бібліотеки, наприклад, Jest, Mocha, Jasmine для юніт-тестування, або Cypress, Selenium для автоматизованого функціонального тестування.

Інструменти для оптимізації інтерфейсу користувача: Для оптимізації завантаження сторінок і покращення користувацького досвіду використовуються інструменти, такі як Webpack (для збирання і пакування ресурсів), Babel (для трансляції сучасного JavaScript у сумісний з усіма браузерами), і лінтери для перевірки стилю коду (наприклад, ESLint).

Ці інструменти допоможуть ефективно розробляти, тестувати і підтримувати веб-сайт на JavaScript.

Порівнюємо JavaScript фреймворки з метою вибору одного для наших задач.

Порівняння JavaScript фреймворків може бути досить об'ємним завданням через велику кількість існуючих фреймворків і бібліотек. Однак я можу навести загальні порівняльні критерії для кількох популярних фреймворків:

### 1. Angular:

Розробник: Google.

Тип: Full-featured фреймворк.

Основні особливості:

Декларативний шаблонізатор.

Dependency Injection (DI) для кращої керованості об'єктами.

Один з основних уявлень — "концепція одноразової прив'язки" (one-way data binding).

### 2. React:

Розробник: Facebook.

Тип: Бібліотека для створення інтерфейсів.

Основні особливості:

Віртуальний DOM для оптимізації швидкодії.

Компонентний підхід до побудови інтерфейсів.

JSX — розширення JavaScript для декларативного опису UI.

### 3. Vue.js:

Розробник: Єван Ю.

Тип: Прогресивний JavaScript фреймворк.

Основні особливості:

Легковаговий, простий у вивченні.

Відмінна документація.

Такі концепції, як "реактивність" і "компонентність", які дозволяють побудову динамічних інтерфейсів.

### 4. Svelte:

Розробник: Rich Harris.

Тип: Компілятор, який генерує відмінений JavaScript.

Основні особливості:

Все компілюється в звичайний JavaScript на етапі збірки.

Основний акцент на ефективність виконання і мінімізацію розміру коду.

Не потребує використання віртуального DOM.

Порівняння за критеріями:

Складність навчання: Vue.js і Svelte вважаються легшими для вивчення порівняно з Angular і React.

Швидкодія: Angular і React мають віртуальний DOM для оптимізації, тоді як Vue.js і Svelte прагнуть до мінімізації втрат часу на рендерінг.

Екосистема: React має найбільшу екосистему, Angular має вбудовану різноманітність інструментів, а Vue.js і Svelte надають простоту використання.

Кожен з цих фреймворків має свої сильні та слабкі сторони, і вибір залежить від конкретних потреб проекту.

Розглянемо більш детально фреймворк React на який пав наш вибір.

## 1.8 Фреймворк React.js

React має кілька значних переваг, які роблять його одним з найпопулярніших фреймворків для створення інтерфейсів у веб-розробці:

**Віртуальний DOM:** React використовує віртуальний DOM для оптимізації процесу оновлення інтерфейсу. Це дозволяє зменшити кількість операцій оновлення реального DOM, що призводить до покращення продуктивності додатка.

**Компонентна архітектура:** React базується на ідеї компонентів, які є самостійними, відокремленими частинами інтерфейсу. Це сприяє полегшенню розподілу обов'язків у розробці, поліпшує повторне використання коду і підтримує чистоту коду.

**JSX:** JSX — це розширення JavaScript, яке дозволяє описувати структуру UI в декларативному стилі прямо в JavaScript коді. Це робить код більш читабельним і підвищує його масштабованість.

**Широкий вибір бібліотек і інструментів:** Велика екосистема React включає в себе багато сторонніх бібліотек і інструментів, що сприяють розширенню можливостей фреймворку і покращенню продуктивності розробників.

Активна спільнота і підтримка: React має велику активну спільноту розробників, яка створює різноманітні розширення, документацію, уроки та відповідає на питання. Це робить вивчення та вирішення проблем більш ефективними.

Гнучкість і масштабованість: React підходить для розробки як малих, так і великих проєктів. Його гнучкість дозволяє використовувати його для створення інтерфейсів різного рівня складності і масштабу.

Усі ці фактори роблять React потужним інструментом для сучасної веб-розробки, здатним забезпечити високу продуктивність та ефективність розробки інтерфейсів.

React.js - це потужна бібліотека для створення інтерфейсів користувача веб-додатків. Ось основні особливості та принципи використання React:

Компонентна модель: Основний принцип React полягає в розділенні користувацького інтерфейсу на невеликі, самостійні компоненти. Кожен компонент відповідає за свою частину інтерфейсу, що сприяє повторному використанню коду, покращує підтримку та чистоту коду.

Віртуальний DOM: React використовує віртуальний DOM для ефективного оновлення інтерфейсу. Замість безпосередньої маніпуляції реальним DOM, React порівнює віртуальний DOM з реальним і визначає мінімальні зміни, які потрібно внести для оновлення інтерфейсу.

JSX: JSX — це розширення JavaScript, яке дозволяє описувати структуру користувацького інтерфейсу в декларативному стилі, що схожий на HTML. Це полегшує розробку інтерфейсу, покращує читабельність коду та сприяє його рефакторингу.

Одноразова прив'язка даних (One-way Data Binding): React використовує одноразову прив'язку даних, що означає, що дані потікають лише в одному напрямку — від батьківського компонента до дочірнього. Це спрощує управління станом додатка і підтримує принципи чистої архітектури.

Декларативний підхід до програмування: React спрощує створення інтерфейсу через декларативний підхід, де розробник описує, як повинен виглядати інтерфейс у різних станах, а не описує кожен крок процесу вручну.

Масштабованість: React підходить як для малих, так і для великих проектів. Він має зручні інструменти для організації коду, такі як контексти, хуки (hooks), портали і т.д., що полегшують роботу з складними структурами додатків.

Екосистема і спільнота: React має велику і активну спільноту розробників, яка створює багато корисних бібліотек, компонентів та інструментів. Це дозволяє швидко знаходити рішення для різних завдань і підтримує живий обмін знаннями.

Використання React забезпечує ефективний і швидкий процес розробки інтерфейсів, спрощує підтримку коду і підвищує його надійність, що робить його відмінним вибором для багатьох веб-додатків. Тому оберемо для наших задач саме його.

## 1.9 Фреймворк Node.js

Для організації бекенду необхідно також обрати фреймворк для цих задач підійде популярний – Node.js.

Node.js є потужною платформою для розробки серверних застосунків, яка базується на JavaScript. Ось основні особливості використання Node.js:

Асинхронний і подієвий модель виконання: Node.js використовує неблокуючий ввід/вивід (non-blocking I/O) та однопотокову подієву модель, що дозволяє ефективно обробляти велику кількість одночасних запитів без блокування і затримок.

JavaScript на сервері: Node.js дозволяє використовувати JavaScript на сервері, що спрощує розробку, оскільки розробникам не потрібно вивчати різні мови програмування для клієнтської та серверної частини.



**Широке використання:** Node.js широко використовується для створення веб-серверів і мікрослужб, RESTful API, реального часу застосунків, чатів, інтернет-магазинів, інструментів для розробки і тестування та інших додатків.

**Потужна екосистема:** Node.js має велику кількість сторонніх бібліотек і пакетів (за допомогою npm - менеджера пакетів Node.js), які значно полегшують розробку. З цими пакетами розробники можуть швидко додавати функціональність до своїх додатків без необхідності писати все з нуля.

**Масштабованість:** Node.js підтримує горизонтальну масштабованість, що дозволяє розширювати додатки на багато серверів.

**Швидкодія:** Завдяки асинхронній моделі виконання і використанню JavaScript V8 engine (що також використовується в браузері Chrome), Node.js забезпечує високу швидкодію виконання коду.

**Спільнота та підтримка:** Node.js має активну спільноту розробників, що розширюється, а також підтримку від головних компаній і організацій (таких як Node.js Foundation, а тепер OpenJS Foundation), що забезпечує стабільність і розвиток платформи.

Використання Node.js дозволяє створювати ефективні і масштабовані серверні додатки, забезпечуючи при цьому високу продуктивність і гнучкість.

## **1.10 Система керування базою даних MongoDB**

Тепер розглянемо більш детально СКБД для наших потреб. Ми зупинили свій вибір на MongoDB. MongoDB — це документ-орієнтована NoSQL база даних, яка відрізняється від традиційних реляційних баз даних. Ось основні особливості використання MongoDB:

**Документ-орієнтована модель даних:** MongoDB зберігає дані у вигляді документів у форматі BSON (Binary JSON). Кожен документ містить ключі і значення, аналогічно JSON об'єкту. Це дозволяє зберігати структуровані, напівструктуровані та навіть неструктуровані дані без потреби визначення схеми.

**Гнучкість і масштабованість:** MongoDB підтримує горизонтальну масштабованість, що означає, що можна легко розширювати базу даних, додаючи нові сервери. Це робить MongoDB ідеальним вибором для високонавантажених систем.

**Вбудована реплікація і шардування:** MongoDB надає можливість створювати репліки (резервні копії) даних для забезпечення високої доступності і забезпечує можливість розподілення даних (шардування) між різними серверами.

**Запити індексів:** MongoDB підтримує створення різних типів індексів (простих, складних, текстових), що підвищує швидкодію запитів і зменшує час виконання запитів до бази даних.

**Агрегації і аналіз даних:** MongoDB має потужні засоби для агрегації даних, що дозволяє виконувати складні операції обробки даних без необхідності переносу їх у зовнішні інструменти аналізу.

**Гнучкий мовний запит:** MongoDB використовує мову запитів, яка нагадує JavaScript (MongoDB Query Language), що полегшує розробникам працювати з базою даних і виконувати різноманітні запити.

**Добре підтримується спільнотою:** MongoDB має активну спільноту розробників і підтримується великою кількістю сторонніх бібліотек, інструментів і сервісів.

**Використання MongoDB** дозволяє ефективно зберігати, керувати і аналізувати великі обсяги даних, забезпечуючи при цьому гнучкість і високу швидкодію роботи з базою даних.

## **1.11 Порівняння MongoDB і MySQL**

Порівняння MongoDB і MySQL можна провести з різних точок зору, оскільки вони представляють собою різні типи баз даних: NoSQL (MongoDB) і реляційну (MySQL). Ось основні аспекти для порівняння:

### **1. Модель даних:**

**MySQL:** Це реляційна база даних, де дані організовані у вигляді таблиць з рядками і стовпцями. Вона використовує SQL для запитів і має строго визначену схему зі зв'язками між таблицями.

**MongoDB:** MongoDB є документ-орієнтованою базою даних, де дані зберігаються у вигляді документів у форматі BSON (Binary JSON). Кожен документ може мати різну структуру, і він зберігає всі дані, що відносяться до об'єкта, у одному документі.

## 2. Мова запитів:

**MySQL:** Використовує мову запитів SQL (Structured Query Language), яка є стандартом для реляційних баз даних. SQL дозволяє виконувати складні запити, агрегації, з'єднання таблиць тощо.

**MongoDB:** Використовує мову запитів, схожу на JavaScript, яка називається MongoDB Query Language. Ця мова спеціалізується на роботі з документами і дозволяє виконувати різноманітні операції з даними, включаючи пошук, агрегації, фільтрації тощо.

## 3. Схема і гнучкість:

**MySQL:** Має строго визначену схему з попередньо визначеними структурами таблиць і зв'язками між ними. Зміни в схемі можуть вимагати перегляду і поновлення даних у відповідних таблицях.

**MongoDB:** Не має фіксованої схеми даних. Кожен документ може мати різну структуру, що дозволяє гнучко працювати з даними без потреби у попередньому визначенні схеми. Це особливо корисно для ситуацій, коли структура даних може часто змінюватися або коли потрібно зберігати різні типи даних в одній колекції.

## 4. Масштабованість:

**MySQL:** Горизонтальна масштабованість MySQL може бути складною, оскільки вимагає реплікації і розподіленого збереження даних для забезпечення високої доступності та масштабованості.

**MongoDB:** MongoDB підтримує легку горизонтальну масштабованість через шардування (розподіл даних на кілька серверів). Це робить MongoDB

ідеальним вибором для високонавантажених систем і додатків, які вимагають розширення обсягів даних.

#### 5. Використання:

**MySQL:** Зазвичай використовується для традиційних веб-додатків, систем управління вмістом, електронної комерції, тощо, де важлива структурованість даних і їх консистентність.

**MongoDB:** Використовується для систем зберігання, обробки та аналізу великих обсягів даних, додатків реального часу, систем моніторингу, аналітики та інших випадків, де важлива гнучкість в роботі з даними.

Обираючи між MongoDB і MySQL, слід враховувати тип даних, які ми опрацюємо, характер додатка, його вимоги до швидкодії, масштабованості і гнучкості. MySQL підходить для традиційних реляційних даних і структурованих операцій, тоді як MongoDB забезпечує більшу гнучкість і масштабованість для сучасних, неструктурованих даних і вимог до обробки великих обсягів інформації. Оскільки в інтернет магазині можуть з часом з'являтися нові категорії товарів тому зручно буде використати саме MongoDB.

### **1.12 Висновок до першого розділу**

В першому розділі кваліфікаційної роботи описані основні технології які використовуються при створенні веб-сайтів з використанням JavaScript. В якості технології управління базою даних було обрано MongoDB оскільки забезпечує більшу гнучкість і масштабованість для сучасних, неструктурованих даних і вимог до обробки великих обсягів інформації.. Оскільки ми обрали JavaScript то необхідно згадати, за фреймворки для написання фронт-енду використаєм React.js, а для написання бек-енду Node.js.

## РОЗДІЛ 2. РОЗРОБКА ВЕБ-САЙТУ

### 2.1 Опис загальних кроків розробки веб-сайту

Розробка веб-сайту з використанням React.js для клієнтської частини, Node.js для серверної частини і MongoDB для бази даних може бути поділена на декілька основних кроків. Нижче наведено загальний огляд процесу розробки:

Кроки розробки веб-сайту з використанням React, Node.js і MongoDB:

Планування і архітектура:

Визначення вимог до функціональності і вигляду сайту.

Проектування структури бази даних MongoDB, включаючи колекції і схеми даних.

Вибір архітектури серверної частини на Node.js (наприклад, використання MVC або REST API).

Налаштування середовища розробки:

Встановлення необхідного програмного забезпечення: Node.js, npm (або yarn), MongoDB.

Налаштування проекту React за допомогою Create React App або аналогічного інструменту.

Розробка клієнтської частини з використанням React:

Створення компонентів React для різних частин інтерфейсу користувача.

Реалізація маршрутизації за допомогою React Router для навігації між сторінками.

Взаємодія з сервером через HTTP запити для отримання і відправлення даних (наприклад, за допомогою Fetch або Axios).

Розробка серверної частини на Node.js:

Створення серверної структури: обробники запитів, маршрутизація, контролери.

Підключення до бази даних MongoDB за допомогою офіційного драйвера або бібліотеки ORM, наприклад Mongoose.

Реалізація логіки бізнес-логіки, обробка запитів і відповідей.

Інтеграція клієнтської і серверної частин:

Взаємодія між клієнтською і серверною частинами через API (REST або GraphQL).

Розробка аутентифікації і авторизації користувачів, якщо потрібно.

Тестування:

Написання тестів для перевірки правильності роботи функціональності.

Запуск тестів і виправлення помилок.

Оптимізація і розгортання:

Оптимізація коду для покращення продуктивності і швидкодії сайту.

Розгортання на віддаленому сервері за допомогою платформи, такої як Heroku, AWS, або іншої.

Налаштування середовища розгортання і бази даних.

Підтримка і розвиток:

Моніторинг і підтримка роботи веб-сайту.

Постійний розвиток і оновлення функціональності на основі змінних вимог користувачів і технологій.

Ці кроки є загальними. Важливо дотримуватися кращих практик і забезпечити належну безпеку і продуктивність веб-сайту під час розробки і після її завершення.

## **2.2 Дизайну веб-сайту та його розробка**

**Дизайн сайту. Кольорова гама.** Традиційно застосуємо білий колір при оформленні веб сайту оскільки такі підходи типові і для інших популярних сайтів інтернет магазину. Зокрема на рисунку 2.1 наведений приклад відомого сайту продажу інструментів «Дніпро-М».

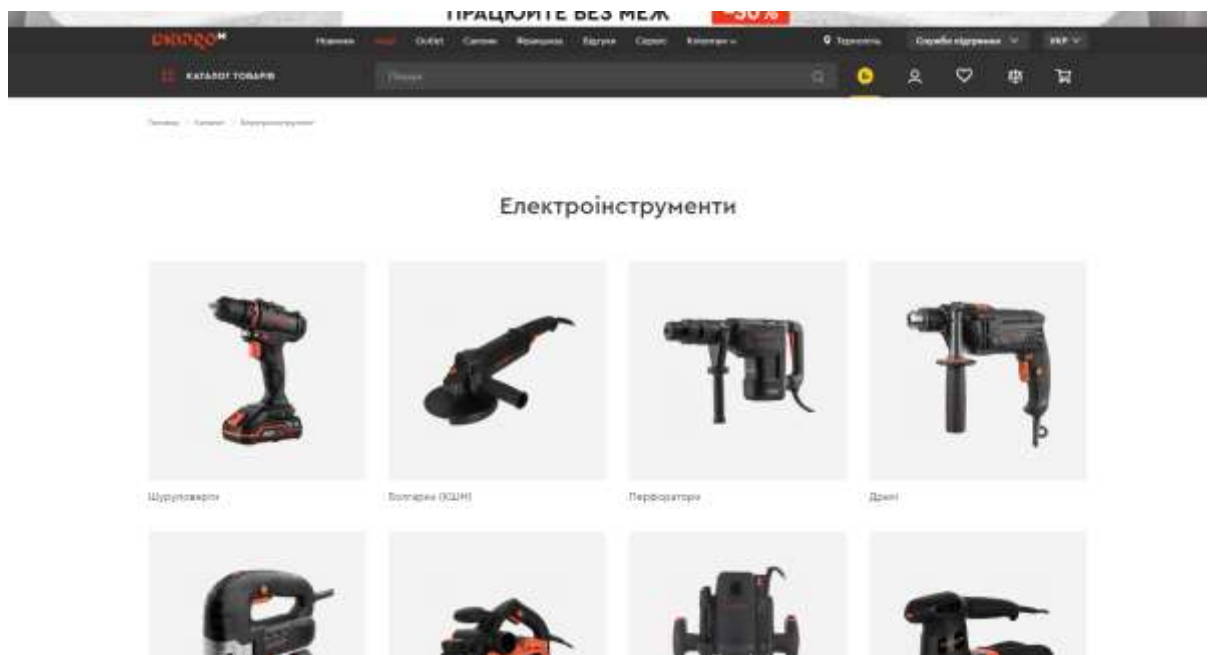


Рисунок 2.1 – Вигляд типового веб-сайту інструментів «Дніпро-М»

Інший приклад відомого сайту «makita.market» наведений на рисунку 2.2

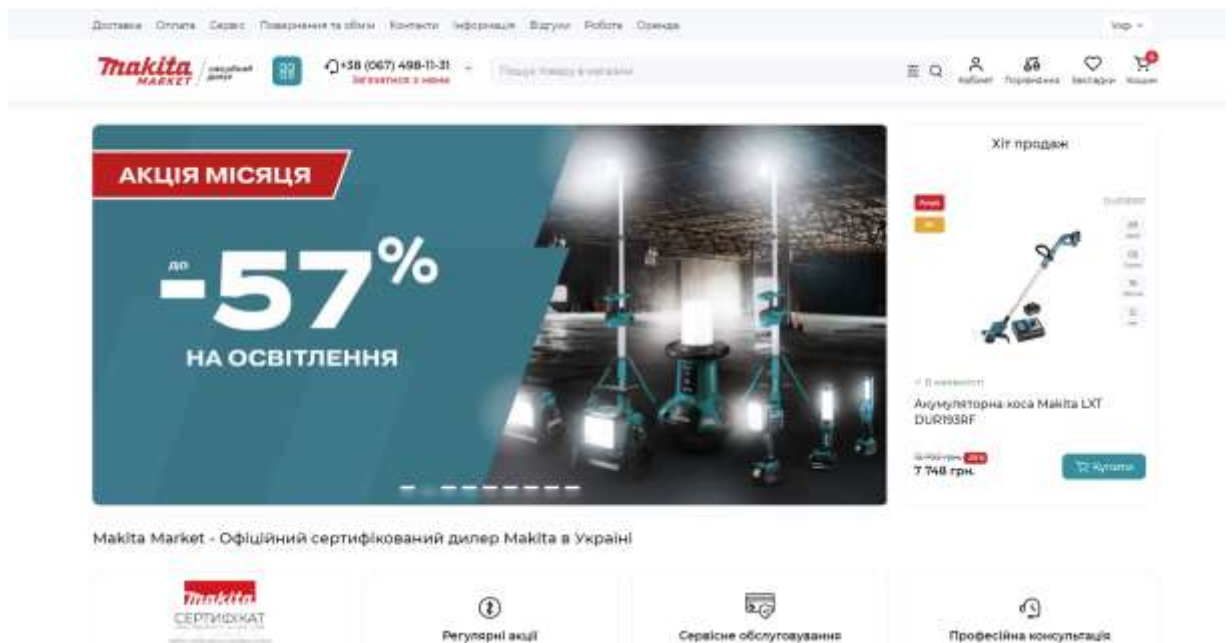


Рисунок 2.2 – Вигляд типового веб-сайту інструментів «makita.market»

Для дизайну нашого сайту також оберемо кольори які не втомлюють очей і дозволяють контрастно відображати інформацію на веб-сайті інтернет магазину. Тому оберемо ці кольори при створенні веб-сайту. При цьому

основні кольори оберемо сірий, білий та для елементів керування помаранчевий.

Зокрема білий фон буде базовим для нашого сайту. Такі елементи як інформаційна стрічка буде відображатися помаранчевим кольором з метою завлїкти увагу користувача. Зокрема будемо відображати інформацію про безкоштовну доставку товару. Елементи такі як логотип телефонів, мова, «улюблені товари», та корзина також подамо у вигляді помаранчевих кольорів.

Якщо говорити про логотип магазину то тут варто не придїляти особливої уваги. Оскільки в подальшому якщо власник магазину буде замовляти конкретний логотип тоді можна буде його детально промальвати. Тому на даному етапі ми обмежимося саме назвою магазину – «PROF».

Сам логотип традиційно розмістимо зліва на сторінці в верхній її частині.

Інше оформлення сайту буде типовим як на популярних веб-сайтах інтернет магазинів зокрема основні заголовкові будуть оформлені чорним кольором на фоні сірого чи білого, що дозволяє чітко видїляти їх. Основний текст чорного та в деяких місцях сірого кольору. Тут необхідно сказати що підсавічені елементи меню чи вибрані елементи меню будуть помаранчевого кольору на білому фоні. Більшість із посилань відображені чорним кольором.

Щн одним елементом є панель пошуку товарів на сайті. Вона оформлена зліва в верху білим кольором на сірому фоні.

Перейдемо до розробки структури сайту та його сторінок.

**Структура сайту.** Важливим елементом при розробці будь-якого веб-сайту є розробка його структури сторінок. Тому даний сайт має так структуру сторінок: головна, каталог, доставка і оплата, про магазин, блог, система знижок, акції, гуртовикам та контакти.

На головній сторінці відображається інформація про продукцію інтернет-магазину «PROF». Зокрема тут представлені товари такі як «Набори інструментів», «Сезонні товари», «Автоінструменти», «Генератори», «Пневмоінструменти та компресори», «Бензоінструмент», «Будівельні інструменти», «Засоби кріплення», «Ручні інструменти», «Все для зварювання»,



«Електроінструмент». Окрім цього на сторінці передбачена можливість вибору товару по категорії: «Бензо», «Електро», «Будівельні», «Ручні», «Садова», «Зварювання». Взаємодія сторінки та структура веб-сайту наведена на рисунку 2.3

2.3

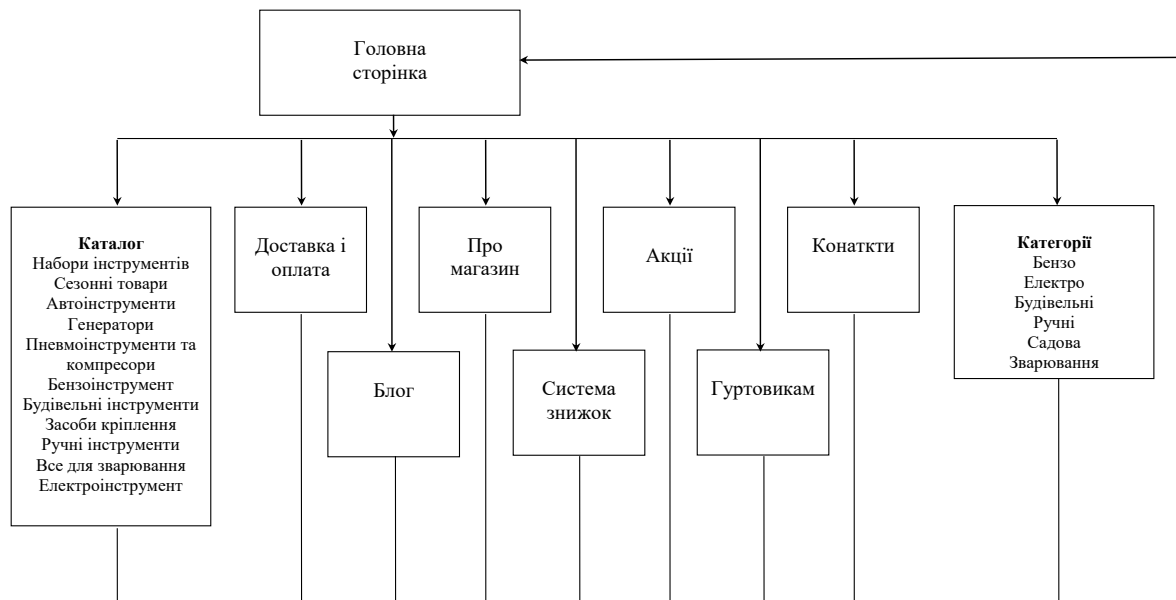


Рисунок 2.3 – Вигляд структури веб-сайту

Головна сторінка веб-сайту інтернет магазину лише ознайомлює клієнта з видами продукції «PROF» і заціклює його. На рисунку 2.4 для прикладу наведена сторінка веб-сайту.

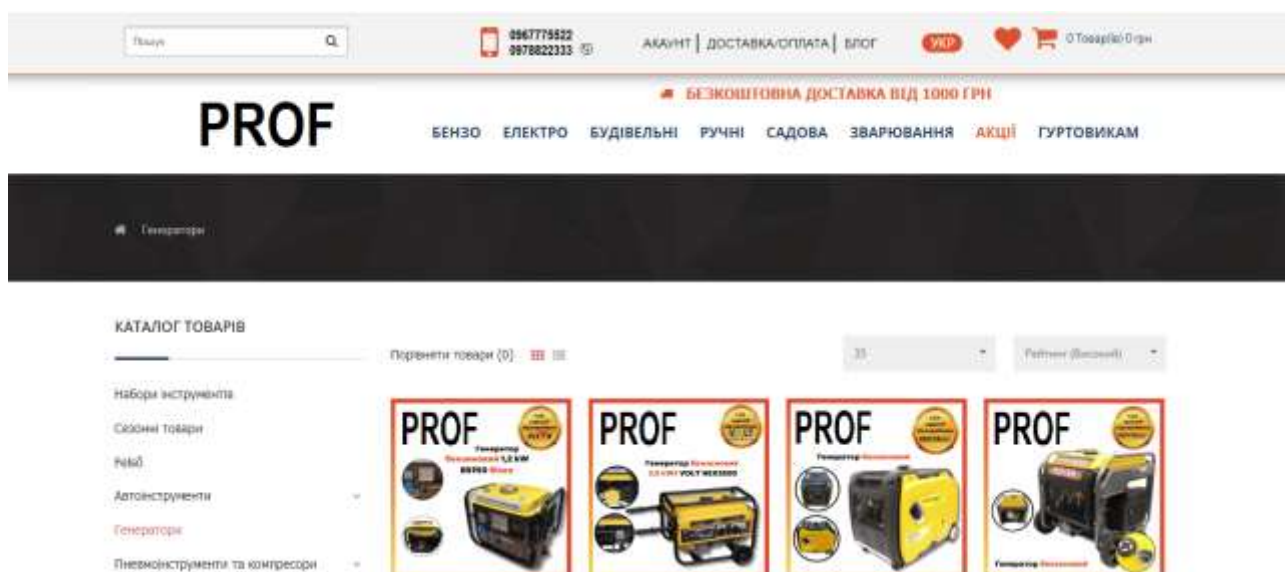


Рисунок 2.4 – Вигляд головної сторінки розробленого веб-сайту

Стоїрнка «Каталог» надає інформацію про весь асортимент продукції магазину інструментів «PROF». Для прикладу на рисунку 2.4 наведена інформація про набори інструментів. Крім цього для зручності на сторінці наведена панель меню в якій можна обрати категорії інструментів такі як «Бензо», «Електро», «Ручні» та інші. Якщо клієнт обрав конкретний вид продукції то її можна додати до кошика. Кількість одиниць у кошику при цьому зміниться на одну. По завершенні вибору товарів можна перейти у скриньку покупок та оформити замовлення на товар і замовити вбраний вид доставки.

При виборі того чи іншого товару можна отримати додаткову проконсультацію у менеджера з продажу для цього є на сторінці телефони чи вайбер контакт.

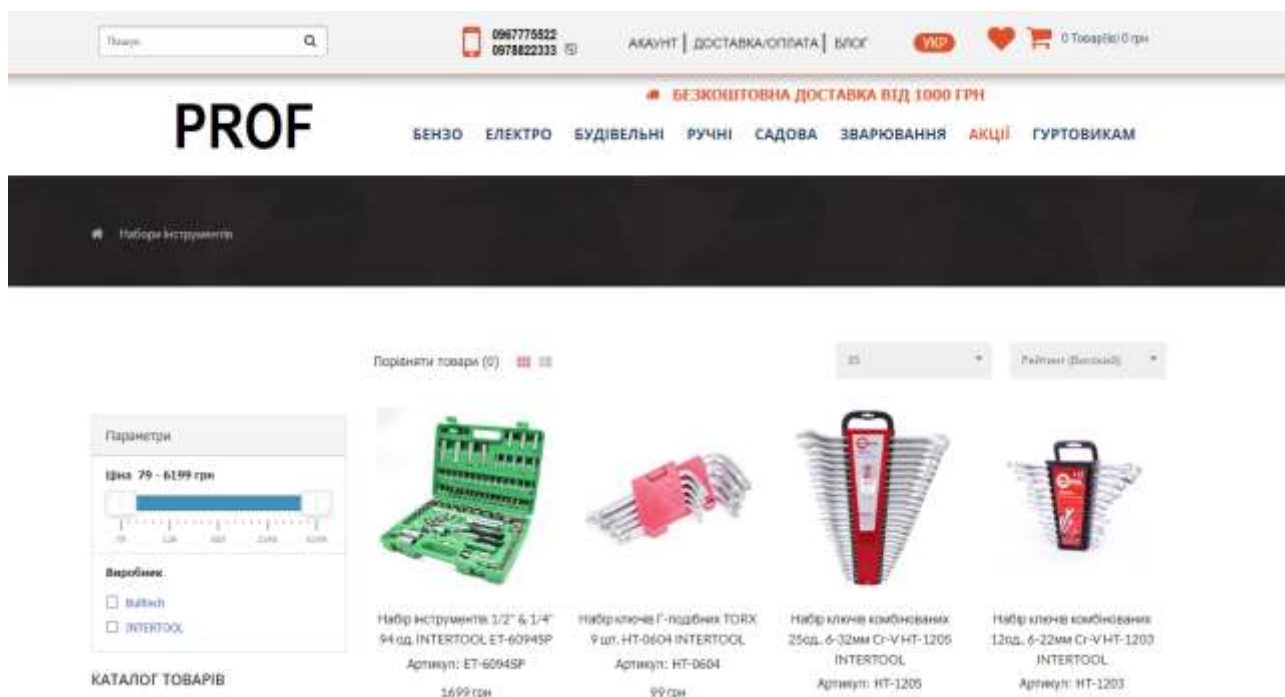


Рисунок 2.5 – Вигляд сторінки «Каталог» розробленого веб-сайту (набір інструментів)

Для того щоб клієнт міг ознайомитись та обрати вид доставки навіть не здійснивши покупку передбачена сторінка «Доставка та оплата».

На сторінці «Доставка та оплата» наведена вся необхідна інформація з якої може бути корисна клієнту магазину. Для прикладу на рисунку 2.6 наведена сторінка веб-сайту «Доставка та оплата».



Рисунок 2.6 – Вигляд сторінки «Доставка та оплата» розробленого веб-сайту

Розглянемо інші сторінки веб-сайту інтернет-магазину.

На сторінці «Про магазин» описана вичерпна інформація про магазин «PROF», коли був заснований, скільки на ринку, яку продукцію продає, які відомі марки продукції є та інше. Згодом мабуть на цій сторінці будуть додані відгуки задоволених клієнтів.

На сторінці «Блог» відображаються події, новини які пов'язані з магазином «PROF».

На сторінці «Система знижок» вказана інформація про можливі знижки і накопичувальну систему балів для постійних клієнтів. Це може бути цікаво майстрам які для роботи закупають інструмент чи розхідні матеріали.

На сторінці «Акції» наведена інформація про діючі в даний період акції. Оскільки це разові акції це також може зацікавити широке коло клієнтів магазину.

На сторінці «Гуртовикам» приведена інформація для оптових закупівль та можливі бонуси від здійснених закупівль партіями товарів.

На сторінці «Контакти» розміщена вся необхідна інформація про сам інтернет-магазин «PROF». Із мамої назви сторінки зрозуміло що тут знаходиться контакти, адреса розташування магазину, телефони, години роботи. Також наведена гугл-мапа щоб клієнту було зрозуміло де розташований територіально в місті магазин. На даному етапі проектування інформація про телефони та адресу коректно не вказана, оскільки це поки розробка інтернет магазину, який можна використати на практиці.

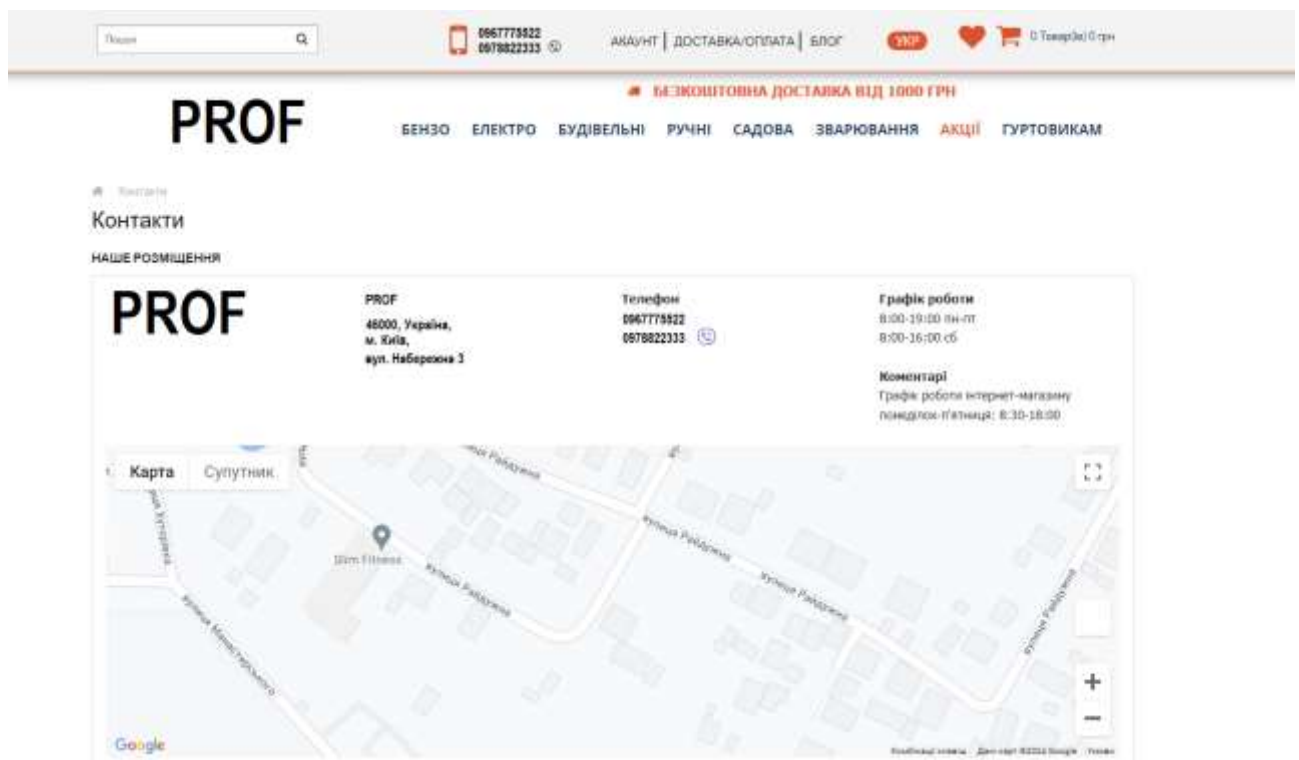


Рисунок 2.7 – Вигляд сторінки «Контакти» розробленого веб-сайту

Для прикладу сторінка «Контакти» подана на рисунку 2.7.

### 2.3 Тестування веб-сайту

Тестування веб-сайтів, написаних на мові JavaScript, включає різні аспекти від перевірки функціональності до візуальної цілісності і

продуктивності. Розглянемо основні види тестування, які можуть бути застосовані до веб-сайтів на JavaScript:

Види тестування веб-сайтів на JavaScript:

Unit-тестування:

Опис: Тестування окремих функцій, методів або модулів коду для перевірки їхньої правильності.

Інструменти: Наприклад, для JavaScript можуть використовуватися бібліотеки тестування, такі як Jest, Mocha, або QUnit.

Інтеграційне тестування:

Опис: Тестування взаємодії між різними компонентами, модулями або сервісами веб-сайту.

Інструменти: Наприклад, для автоматизації інтеграційних тестів можна використовувати фреймворки, такі як Selenium або Puppeteer.

Функціональне тестування:

Опис: Тестування функціональних вимог веб-сайту, перевірка правильності реакції на користувацькі взаємодії і введення.

Інструменти: Можна використовувати фреймворки для UI тестування, такі як Cypress або TestCafe.

Візуальне тестування:

Опис: Перевірка зовнішнього вигляду і візуальної цілісності сторінок веб-сайту на різних пристроях і браузерах.

Інструменти: Наприклад, фреймворки для візуального тестування, такі як Applitools або Percy.

Видимість в пошукових системах (SEO тестування):

Опис: Перевірка, як сайт оптимізований для пошукових систем, включаючи правильність мета-тегів, URL-адрес, швидкість завантаження сторінок і т.д.

Інструменти: Можуть використовуватися онлайн-інструменти для аналізу SEO, такі як Google PageSpeed Insights або SEMrush.

Швидкість та продуктивність:

Опис: Вимірювання часу завантаження сторінок, продуктивності JavaScript коду і оптимізації для швидкості.

Інструменти: Інструменти для профілювання коду, наприклад, Chrome DevTools або Lighthouse.

Безпека:

Опис: Тестування на вразливості безпеки, такі як XSS (міжсайтовий скриптинг) або CSRF (фальшивий запит між сайтами).

Інструменти: Засоби для сканування на вразливості, наприклад, OWASP ZAP або Burp Suite.

Адаптивний дизайн і сумісність:

Опис: Перевірка відповідності дизайну сторінок на різних пристроях і браузерах.

Інструменти: Ручні тести або використання сервісів для перевірки адаптивності, таких як BrowserStack або CrossBrowserTesting.

Загальні підходи до тестування:

Автоматизація тестів: Автоматизація дозволяє швидко виявляти помилки і забезпечувати стабільність веб-сайту під час розробки і після випуску.

Континуальна інтеграція і доставка (CI/CD): Використання CI/CD підходу дозволяє автоматизувати процеси тестування, розгортання і відстеження змін в коді.

Залежно від потреб і особливостей проекту, вибір конкретних видів тестування і інструментів може відрізнятись. Важливо створити комплексну стратегію тестування, яка забезпечить високу якість і надійність веб-сайту. Тому ми оберемо саме Selenium для тестування розробленого веб-сайту.

## **2.4 Висновок до другого розділу**

В другому розділі кваліфікаційної роботи описані основні кроки по розробці веб-сайту його структура, основні сторінки, дизайн та кольорова

гамма. Тут описано чому саме обрано таку структуру та дизайн веб-сайту. Описані основні його сторінки та структурні елементи. Проведено тестування розробленого веб-сайту у різних броузерах для цього використано інструмент Selenium які підтвердили коректне відображення всіх необхідних елементів розробленого веб-сайту.

## РОЗДІЛ 3. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

**3.1 Безпека життєдіяльності (БЖД) наука, що вивчає проблеми безпечного перебування людини в довкіллі в процесі різних видів її діяльності.**

В центрі уваги курсу БЖД – людина як самоціль розвитку суспільства, а його лейтмотивом є вислів старогрецького філософа Протагора “Людина є мірило всіх речей”. Людина має цінність не тільки як робоча сила, яку потрібно охороняти в процесі трудової діяльності (чим з певним успіхом і займалася охорона праці), а як неповторний індивідуум, який зберігає свою цінність незалежно від свого конкретного місцезнаходження і виконуваних функцій, признаних суспільством необхідними і корисними (відпочинок, побут, заняття спортом, виконання громадських обов’язків тощо).

Безпека – це стан діяльності, при якому з певною ймовірністю виключається прояв небезпек. Безпека – це мета, а безпека життєдіяльності – засоби, шляхи, методи її досягнення.

Актуальність дисципліни ще більше зростає у зв’язку з існуванням аксіоми про потенційну небезпеку діяльності: в жодному виді діяльності неможливо досягнути абсолютної безпеки, будь-яка діяльність потенційно небезпечна.

Завдання БЖД є розробка методів прогнозування, вивчення та ідентифікації шкідливих факторів, їх впливу на людину і довкілля.

Курс БЖД призначений:

- сприяти усвідомленню, що в центрі уваги повинна бути людина, як головна цінність суспільства, та виховати в людині гуманне,
- свідоме ставлення до питань особистої безпеки та безпеки оточуючих в усіх сферах відносин;
- виробити навички ідентифікації небезпечних та шкідливих факторів і створення сприятливих умов життєдіяльності людей на певній території;



- тримати на контролі проектування нової техніки і технологічних процесів згідно з сучасними вимогами екології і з урахуванням стійкості функціонування господарських об'єктів та технічних систем;

- прогнозувати можливу обстановку і приймати грамотні рішення в умовах надзвичайних ситуацій щодо захисту населення та персоналу об'єктів від можливих негативних наслідків;

- забезпечити якісне засвоєння нового стереотипу поведінки людини з метою виживання в нових природних та антропогенних умовах.

Завдання курсу “Безпека життєдіяльності” (БЖД) полягає у чіткому розумінні небезпечних чинників у ситуаціях, що виникають як у середовищі проживання людини, так і у середовищі навчання і праці.

БЖД – це ступінь захисту людини від надзвичайної небезпеки, де під терміном “небезпека” мається на увазі вплив на людину факторів, які можуть викликати відхилення стану її здоров'я від нормального. Природа цих факторів може бути пов'язана як з причинами природного або соціально-економічного характеру (екологічними катастрофами, низьким економічним рівнем життя та ін.), так і з причинами техногенного характеру (з рівнем забруднення навколишнього середовища як наслідком виробничої діяльності людини, аваріями, катастрофами на підприємстві, транспорті, війнами та ін.).

Зміни, що відбувається в нашій країні сьогодні: приватні підприємства, приватизація, кооперація, ринкові відносини, інфляція і т.д., здатні загострити проблеми безпеки – життєдіяльності людини. Так, зараз скорочуються наукові дослідження з проблем безпеки, зменшується чисельність інспекторів, скорочується служба охорони праці. Надзвичайні ситуації в Україні виникають у 5-8 разів частіше, ніж в інших промислово розвинутих країнах світу, в результаті чого щорічно гине понад 1% населення (більше 50 тисяч осіб). Майже третина аварій і нещасних випадків на виробництві пов'язана з незадовільним знанням людей порядку дій у надзвичайних ситуаціях. Це – наслідок серйозних недоліків у навчанні як на виробництві, так і в освітніх закладах. Вивчення предмету БЖД є важливим кроком у розширенні світогляду

майбутніх фахівців про безпеку людини та її захист у процесі виробничої і побутової діяльності, в умовах надзвичайних ситуацій мирного та воєнного часу.

### **3.2 Основні небезпеки в урбанізованому середовищі**

Сучасній людській цивілізації властиві стрімкі темпи урбанізації. Вони вирішально зумовлені двома факторами – “демографічним вибухом” другої половини ХХ ст. та науково-технічною революцією в усіх сферах.

Урбанізація (від лат. *urbanos* – міський) означає процес зростання міст і міського населення та підвищення їх ролі в соціально-економічному та культурному житті суспільства. Способи виникнення міст в історії людства були різними. Міста виникали як сумісні поселення ремісників, що полегшувало їх виробничу діяльність, як центри торгівлі, як воєнні укріплення (фортеці).

Процес світової урбанізації розпочався в Європі, де вперше почали формуватися міста завдяки концентрації засобів виробництва, великої промисловості. Становлення урбанізації почалося на зламі XVII-XIX ст., коли міста в Західній Європі зосереджували найважливіші засоби виробництва й посідали ключові позиції у світовій економіці. Локальний розвиток міст породив певну просторову послідовність світового процесу урбанізації: Західна Європа – Північна Америка – Австралія та Океанія – Східна Європа – Латинська Америка – Азія – Африка.

Уже сьогодні в багатьох країнах світу, особливо економічно розвинених, частка міського населення становить 85-90% і більше в загальній його чисельності. За прогнозами Комісії ООН з народонаселення, на початку ХХІ ст. у містах буде мешкати не менше 51% усього населення земної кулі, тоді як 1970 р. частка міського населення становила 38,6%. З'являються і зростають багатомільйонні міста – мегаполіси (Нью-Йорк, Лондон, Токіо та ін.), збільшується їх кількість, розміри та проблеми.

Проявився процес урбанізації і в Україні. До 1918 р. країна була аграрною і в містах проживало 18% населення. Інтенсивна урбанізація в Україні розпочалася в 1926-1939 рр., коли було взято курс на індустріалізацію народного господарства, тобто всього лише за 13 років чисельність міського населення зросла у 2,4 рази. За 1940-70 рр. чисельність міського населення в Україні зростала вже значно нижчими темпами, бо за 30 років воно збільшилось у 1,9 рази. З середини 50-х років ХХ ст. почався новий етап інтенсивного зростання кількості міст і чисельності міського населення в Україні. Тільки за останні 30 років частка міського населення в Україні зросла у 2,2 рази та становила на кінець ХХ ст. близько 70% загальної чисельності населення. За кількістю великих міст (з населення понад 100 тис.) наша держава тепер посідає одне з провідних місць серед країн світу, таких міст зара – 61. В Україні є 7 міст з населенням, яке перевищило або майже сягає мільйона осіб: Київ, Дніпропетровськ, Одеса, Донецьк, Харків, Запоріжжя та Кривий Ріг. До речі, для Києва останні 30 років був характерний надзвичайно інтенсивний демографічний розвиток: чисельність його населення зросла майже у 2,5 рази і досягла на 2000 рік 2 млн. 600 тисяч мешканців.

Урбанізацію неможливо розглядати без зв'язку з розвитком суспільного виробництва, зокрема важкої індустрії, енергетики, хімічної промисловості тощо. Разом з нарощуванням промислового потенціалу, створенням нових галузей виробництва у великих містах зростає чисельність населення. Сучасні великі міста – це центри зосередження багатогалузевої промислової, розгалуженої транспортної мережі в густо населених житлових масивах. Причому найважливішим джерелом зростання міського населення була й все ще залишається міграція сільських жителів у міста. На неї припадає більше половини приросту міського населення в Україні.

Сучасне місто надає своїм жителям багато переваг економічного, соціального та суб'єктивного характеру, а саме:

- наявність місць роботи та можливість зміни роботи;
- зосередження закладів науки та культури;

- забезпечення висококваліфікованої медичної допомоги;
- можливість створювати кращі житлові та соціально-побутові умови життя;
- розвиток міжнародного та регіональної культури.

Незважаючи на переваги міського життя, міське середовище для людей є штучним і відірваним від природного, того, в якому тисячоліттями проходило їхнє життя. Штучне міське середовище шкідливо впливає на здоров'я населення через забруднення атмосферного повітря, дефіцит сонячного проміння, води, а також скупченістю населення, недостатністю зелених насаджень тощо. Також небезпеку для здоров'я людей у місті становлять шумові, вібраційні навантаження, транспортні проблеми, вплив електричних, магнітних, іонізаційних полів.

Отже, в умовах великого міста загострюються всі сторони життєзабезпечення людей: постачання достатньої кількості повноцінних продуктів харчування та питної води; контроль і запобігання забруднення повітря; водних ресурсів, ґрунтів; утилізація та захоронення нагромаджуваних шкідливих виробничих та побутових відходів, а також соціальні проблеми, пов'язані з різким зменшенням вільного “життєвого” простору, зростанням міст у високу, збільшенням захворювань, зумовлених забрудненням та інші.

### **3.3 Загальні закономірності виникнення небезпек**

1. У наш час людина здатна полетіти на Місяць, ми багато знаємо про інші планети, але сили природи нашої власної планети все ще нами не підкорені. У наш цивілізований, технічно розвинений час людство залишається залежним від природних явищ, які досить часто мають катастрофічний характер. Виверження вулканів, землетруси, посухи, селеві потоки, снігові лавини, повені спричиняють загибель багатьох тисяч людей, завдають величезних матеріальних збитків.

Найбільші збитки з усіх стихійних лих спричиняють повені (40%), на другому місці – тропічні циклони (20%), на третьому і четвертому місцях (по 15%) – землетруси та посухи.

Стихійні лиха – це природні явища, які мають надзвичайний характер та призводять до порушення нормальної діяльності населення, загибелі людей, руйнування і нищення матеріальних цінностей.

За причиною виникнення стихійні лиха поділяють на:

тектонічні (пов'язані з процесами, які відбуваються в надрах землі), до них належать землетруси, виверження вулканів;

топологічні (пов'язані з процесами, які відбуваються на поверхні землі), до них належать повені, зсуви, селі;

метеорологічні (пов'язані з процесами, які відбуваються в атмосфері), до них належать спека, урагани, посуха та ін.

а) Тектонічні стихійні лиха

Виверження вулканів. За руйнівною дією та кількістю енергії, яка виділяється при виверженні вулкана, саме це стихійне лихо належить до найнебезпечніших для життєдіяльності людства. Під попелом та лавою гинули цілі міста.

Землетруси. Щорічно вчені фіксують близько 1 млн. сейсмічних і мікросейсмічних коливань, 100 тис. з яких відчуваються людьми та 1000 спричиняють значні збитки.

Землетрус – це сильні коливання земної кори, викликані тектонічними причинами, які призводять до руйнування споруд, пожеж та людських жертв.

Гіпоцентр, або осередок землетрусу, - місце, де зсуваються гірські породи.

Епіцентр – точка на поверхні землі, що знаходиться прямо над гіпоцентром.

Основними характеристиками землетрусів є: глибина осередку, магнітуда, інтенсивність енергії на поверхні землі.

## Ознаки близького землетрусу

Таблиця 3.1

Ознаки близького землетрусу	- запах газу, де раніше цього не відзначалось
	- тривога птахів та домашніх тварин
	- іскри між близько розташованими електричними дротами
	- голубе освітлення внутрішньої поверхні будинків

Першість за кількістю землетрусів утримують Японія та Чілі: понад 1000 в рік, або 3 на день.

Деякі рекомендації правил поведінки в умовах небезпеки землетрусу

При землетрусі ґрунт відчутно коливається відносно недовгий час – тільки декілька секунд, найдовше – хвилину при дуже сильному землетрусі. Ці коливання неприємні, можуть викликати переляк. Тому дуже важливо зберігати спокій. Якщо відчувається здригання ґрунту чи будинку, слід реагувати негайно, пам'ятаючи, що найбільш небезпечні є предмети, які падають.

Перебуваючи у приміщенні, слід негайно зайняти безпечне місце. Це отвори капітальних внутрішніх стін (наприклад, відчинити двері з квартири), кути, утворені ними. Можна заховатися під балками каркасу, під несучими колонами, біля внутрішньої капітальної стіни, під ліжком чи столом. Слід пам'ятати, що найчастіше завалюються зовнішні стіни будинків. Необхідно триматися подалі від вікон та важких предметів, які можуть перекинутися чи зрушити з місця.

Не слід вибігати з будинку, оскільки уламки, які падають уздовж стін, є серйозною небезпекою. Безпечніше перечекати поштовх там, де він вас застав, і, лише дочекавшись його закінчення, перейти у безпечне місце.

Перебуваючи всередині багатоповерхового будинку, не поспішайте до ліфтів чи сходів. Сходові прольоти та ліфти часто обвалюються під час землетрусу.

Після припинення поштовхів потрібно терміново вийти на вулицю, відійти від будівель на відкрите місце, щоб уникнути ударів уламків, які падають.

Перебуваючи в автомобілі, що рухається, слід повільно загальмувати подалі від високих будинків, мостів чи естакад. Необхідно залишатись в машині до припинення поштовхів.

Опинившись в завалі, слід спокійно оцінити становище, надати собі першу допомогу, якщо вона потрібна. Необхідно надати допомогу тим, хто її потребує. Важливо подбати про встановлення зв'язку з тими, хто перебуває зовні завалу (голосом, стуком), Людина може зберігати життєздатність (без води і їжі) понад два тижні.

#### б) Топологічні стихійні лиха

Повені. За даними ЮНЕСКО, від повеней у ХХ ст. загинуло 9 млн. осіб. Недарма в народі кажуть, що найстрашніші для людини це вода і вогонь.

Повінь – це значне затоплення місцевості внаслідок підйому рівня води в річці, озері, водосховищі, спричинене зливами, весняним таненням снігу, вітровим нагоном води, руйнуванням дамб, гребель тощо. Повені завдають великої матеріальної шкоди та призводять до людських жертв.

#### Наслідки повеней

- затоплення шаром води значної площі землі;
- ушкодження та руйнування будівель та споруд;
- ушкодження автомобільних шляхів та залізниць;
- руйнування обладнання та комунікацій, меліоративних систем;
- загибель свійських тварин та знищення врожаю сільськогосподарських культур;
- вимивання родючого шару ґрунту;
- псування та нищення сировини, паливо, продуктів харчування, добрив тощо;
- загроза інфекційних захворювань (епідемії);
- погіршення якості питної води;

– загибель людей.

Повені відрізняються від інших стихійних лих тим, що деякою мірою прогножуються. Але прогнозувати ймовірність повені набагато легше, ніж передбачити момент її початку. Точність прогнозу зростає при отриманні надійної інформації про кількість та інтенсивність опадів, рівні води в річці, запаси води в сніговому покриві, зміни температури повітря. довгострокові прогнози погоди тощо.

Основний напрям боротьби з повенями в зменшенні максимальних витрат води в річці завдяки перерозподілу стоку в часі (насадження лісозахисних смуг, оранка ґрунту поперек схилу, збереження узбережних смуг рослинності, терасування схилів тощо). Для середніх та великих річок досить дійовим засобом є регулювання паводкового стоку за допомогою водосховищ. Окрім того, для захисту від повеней широко застосовується давно відомий спосіб – влаштування дамб. Для ліквідації небезпеки утворення затворів проводиться розчищення та заглиблення окремих ділянок русла ріки, а також руйнування криги вибухами за 10-15 днів до початку льодоходу.

### **3.4 Висновок до третього розділу**

В третьому розділі кваліфікаційної роботи описано чим займається наука безпека життєдіяльності. Були описані основні небезпеки в урбанізованому середовищі. Описані загальні закономірності виникнення небезпек, зокрема розглянуті різні стихійні лиха, які поділяються на тектонічні стихійні лиха, топологічні стихійні лиха. Розглянуті які можуть бути наслідки повеней та ознаки близького землетрусу.



## ВИСНОВКИ

В даній кваліфікаційній роботі розглянуто основні поняття та технологій веб-програмування, акцент зроблено на JavaScript, розглянуті різні фреймворки, для розробки веб-сайту.

При створенні веб-сайту було опрацьовано наступні задачі:

- Було проаналізовано подібні веб-сайти зокрема Дніпром-М та Макіта;
- Було проаналізовано технологій JavaScript та фреймворки які використовують при розробці веб-сайтів, було обрано фрейворк React.js (фронт-енд), Node.js (бек-енд), систему керування базою даних MongoDB;
- Було проведено розробку дизайну веб-сайту та його структури;
- Було здійснено розробку сторінок інтернет-магазину та необхідних елементів керування;
- Було проведено тестування веб-сайту інтернет-магазину.

Веб-сайт інтернет-магазину, може бути розміщений у мережі Інтернет.

Якщо його наповнити та проводити адміністрування він стане незамінним інструментом для торгівлі через інтернет за межами міста, що дозволить підвищити конкурентно-спроможність магазину. У подальших кроках роботи над веб-сайтом можна змінювати елементи меню кольорову гамму та взагальному його дизайну.

У розділі «Безпека життєдіяльності, основи охорони праці» було описано чим займається наука безпека життєдіяльності. Були описані основні небезпеки в урбанізованому середовищі. Описані загальні закономірності виникнення небезпек, зокрема розглянуті різні стихійні лиха, які поділяються на тектонічні стихійні лиха, топологічні стихійні лиха. Розглянуті які можуть бути наслідки повеней та ознаки близького землетрусу.

## ПЕРЕЛІК ДЖЕРЕЛ

1. Крис Джамса, Конрад Кинг, Энди Андерсон. Креативный Web-дизайн. HTML, XHTML, CSS, JavaScript, PHP, ASP, ActiveX. Текст, графика, звук и анимация. Пер. с англ. – М: ООО “ДиаСофтЮП”, 2005. – 672 с.
2. Закас Ніколас. JavaScript для професіоналів. — Київ: Видавничий дім "Букрек", 2018. — 752 с.
3. Stefanov, S. (2016). React: Up & Running: Building Web Applications. O'Reilly Media. ISBN: 978-1491931820.
4. Banks, A., Porcello, E. (2017). Learning React: Functional Web Development with React and Redux. O'Reilly Media. ISBN: 978-1491954621.
5. Freeman, A. (2019). Pro React 16. Apress. ISBN: 978-1484244508.
6. Fedosejev, A. (2017). React.js Essentials. Packt Publishing. ISBN: 978-1783551620.
7. Bertoli, M. (2017). React Design Patterns and Best Practices. Packt Publishing. ISBN: 978-1786464538.
8. Дакетт Джон. JavaScript і jQuery: Інтерактивна веб-розробка. — Київ: Видавництво "Діалектика", 2019. — 544 с.
9. Браун Етан. JavaScript: Підручник з програмування. — Київ: Видавничий дім "Альтерпрес", 2020. — 416 с.
10. Флегель Девід. JavaScript. Повний курс. — Київ: Видавництво "Кодекс", 2017. — 608 с.
11. Морган Нік. JavaScript для дітей. — Київ: Видавництво "Комп'ютерна академія", 2021. — 160 с.
12. В.А. Гайченко, Г.М. Коваль : “Основи БЖД людини” : Навч. посіб. – 2-ге вид. стереотип. – К.: МАУП, 2004.-232с.: іл.-Бібліогр.: с. 225-226
13. Джигрей В.С., Жидецький В.Ц. “Безпека життєдіяльності”. Підручник.- Вид. 4-те, доповнене. – Львів: Афіша, 2001. – 256с.
14. Скобло Ю. С, Соколовська Т. Б., Мазоренко Д. І., Тіщенко Л. М., Троянов М. М. – Безпека життєдіяльності: Навчальний посібник для вищих навчальних закладів III – IV рівнів акредитації. – Київ: Кондор, 2003. – 424с.

15. Желібо Є.П., Заверуха Н.М., Зацарний В.В. : “Безпека життєдіяльності” : Навчальний посібник для студентів вищих закладів освіти України I-IV рівнів акредитації / За ред. Є.П. Желібо, В.М. Пічі. – Львів : Піча Ю.В., К.: “Каравела”, Львів:, Новий світ – 2000, 2002, - 328с.

16. Конспект лекцій з дисципліни «Основи безпеки життєдіяльності».

17. Конспект лекцій з дисципліни «Web-програмування та web-дизайн».

18. Пам’ятка-інструктаж з питань охорони праці для студентів Тернопільського державного технічного університету імені Івана Пулюя. – Тернопіль.

19. Hughes-Croucher, T., Wilson, M. (2012). Node: Up and Running: Scalable Server-Side Code with JavaScript. O'Reilly Media. ISBN: 978-1449398583.

20. Cantelon, M., Harter, M., Rajlich, T., Osmani, A. (2013). Smashing Node.js: JavaScript Everywhere. Wiley. ISBN: 978-1119962595.

21. Mead, A. (2019). The Complete Node.js Developer Course (3rd Edition). Udemy Course.

22. Casares, M. (2014). Mastering Node.js. Packt Publishing. ISBN: 978-1782166320.

23. Hapi.js Team. (2015). Hapi.js in Action. Manning Publications. ISBN: 978-1617292735.

24. Methodology of the Formation of Sports Matches Statistical Information Using Neural Networks Sorokivska, O., Lytvynenko, I., Sorokivskyi, O., Kozbur, H., Strutynska, I. CEUR Workshop Proceedings, 2023, 3628, pp. 389–403

25. Information Technologies Use in the Study of Functional Properties of Wheeled Vehicles Lyashuk, O., Stashkiv, M., Lytvynenko, I., Sakhno, V., Khoroshun, R. CEUR Workshop Proceedings, 2023, 3628, pp. 370–381

26. Software for statistical processing and modeling of a set of synchronously registered cardio signals of different physical nature Lupenko, S., Lytvynenko, I., Sverstiuk, A., Horkunenko, A., Shelestovskyi, B. CEUR Workshop Proceedings, 2021, 2864, pp. 194–205

27. Я.В. Литвиненко, М.В. Ілько. Розробка програмного забезпечення з використанням нейронної мережі в задачах прогнозування електронавантажень. // Тези доповідей ІХ науково-технічної конференції «Інформаційні моделі, системи та технології». 2021. – С. 45-46.

28. П.А. Ониськів, Я.В. Литвиненко. Аналіз програмного забезпечення для покращення якості вхідних зображень. Матеріали міжнародної наукової конференції „Іван Пулюй: життя в ім’я науки та України “(до 175-ліття від дня народження). 2020 року, Тернопіль, Україна 2020. ФОП Паляниця ВА. С. 90

29. Конспект лекцій з дисципліни “Основи технічної творчості та наукових досліджень” (частина 1) для здобувачів освітнього ступеня «бакалавр», спеціальностей: 122 «Комп’ютерні науки», 126 «Інформаційні системи та технології», всіх форм навчання. Укладач Я.В. Литвиненко: ТНТУ, 2020 р.

30. Конспект лекцій з дисципліни “Основи технічної творчості та наукових досліджень” (частина 2) для здобувачів освітнього ступеня «бакалавр», спеціальностей: 122 «Комп’ютерні науки», 126 «Інформаційні системи та технології», всіх форм навчання. Укладач Я.В. Литвиненко: ТНТУ, 2020 р.

31. Chodorow, K. (2013). MongoDB: The Definitive Guide. O'Reilly Media. ISBN: 978-1449344689.

32. Plugge, E., Hawkins, T., Membrey, P. (2013). MongoDB Basics. Apress. ISBN: 978-1430258216.

33. Dirolf, D., Wilson, D., Lafrance, N. (2016). MongoDB in Action (2nd Edition). Manning Publications. ISBN: 978-1617291608.

34. Banker, K., Panchal, A., Sarang, M. (2017). MongoDB Applied Design Patterns. Packt Publishing. ISBN: 978-1785289989.

35. Chodorow, K. (2013). Scaling MongoDB. O'Reilly Media. ISBN: 978-1449344238.

# ДОДАТКИ

### Приклад головної сторінки розробленого веб-сайту

```

<!DOCTYPE html>
<!--[if IE]><![endif]-->
<!--[if IE 8 ]><html dir="ltr" lang="uk" class="ie8"><![endif]-->
<!--[if IE 9 ]><html dir="ltr" lang="uk" class="ie9"><![endif]-->
<!--[if (gt IE 9)!!(IE)]><!-->
<html dir="ltr" lang="uk" rel="alternate" hreflang="uk-UA"
typeOf="http://ogp.me/ns#website">
<!--<![endif]-->
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Інтернет-магазин інструментів PROF</title>

<meta property="og: site_name" content ="Інтернет-магазин інструментів
PROF">
<meta property="og:title" content="Інтернет-магазин інструментів PROF"
/><!-- //-->
<base href="https://prof.ua/" />
<meta name="description" content="Інтернет-магазин інструментів
PROF - великий вибір електроінструментів, бензоінструментів, садової
техніки, зварювального обладнання, інструментів для будівництва, а також
прокат інструментів" />
<meta property="og:description" content=" Інтернет-магазин інструментів
PROF - великий вибір електроінструментів, бензоінструментів, садової
техніки, зварювального обладнання, інструментів для будівництва, а також
прокат інструментів" /><!-- //-->
<meta name="keywords" content= "електроінструменти, бензоінструменти,
садова техніка, інструменти для будівництва, зварювальне обладнання,
ручні інструменти, засоби кріплення, автотовари, прокат інструментів" />
<!-- //-->
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
integrity="sha384-
50oBUHEmvpQ+11W4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJ
K28anvf" crossorigin="anonymous">
<link href="https://prof.ua/" rel="canonical" />

```

```

<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
integrity="sha384-
50oBUHEmvpQ+11W4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJ
K28anvf" crossorigin="anonymous">
<link href="https://prof.ua/image/catalog/banner/logoNew_favicon.png"
rel="icon" />
<link href="catalog/view/javascript/bossthemess/bootstrap/css/bootstrap.css"
rel="stylesheet" media="screen" />
<link href="catalog/view/theme/bt_claudine/stylesheet/stylesheet.css"
rel="stylesheet">
<link rel="stylesheet" type="text/css"
href="catalog/view/theme/bt_claudine/stylesheet/bossthemess/bt_stylesheet.css"
/>

<!-- byRa -->
  <!--Inneti JS-->

  <!-- <script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/1.6.2/jquery.min.js"></script>
-->
  <!-- Подключаем jQuery библиотеку -->
  <script type="text/javascript" src="https://code.jquery.com/jquery-
latest.min.js"></script>

<script src="catalog/view/javascript/disableCopy.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/jquery/jquery-2.1.1.min.js"
type="text/javascript"></script>
<script type="text/javascript"
src="catalog/view/javascript/bossthemess/jquery.smoothscroll.js"></script>
<script src="catalog/view/javascript/bossthemess/jquery.jgrowl.js"
type="text/javascript"></script>

<script src="catalog/view/javascript/bossthemess/bootstrap/js/bootstrap.min.js"
type="text/javascript"></script>
<!-- <link rel="stylesheet" type="text/css"
href="catalog/view/theme/bt_claudine/stylesheet/modal-contact-form.css" /> -->
<!-- <link rel="stylesheet" type="text/css"
href="catalog/view/theme/bt_claudine/stylesheet/bossthemess/responsive.css" />
-->
<link rel="stylesheet" type="text/css"
href="catalog/view/theme/bt_claudine/stylesheet/bossthemess/cs.animate.css" />

```

```

<link rel="stylesheet" type="text/css"
href="catalog/view/theme/bt_claudine/stylesheet/bossthemes/jquery.jgrowl.css"
/>
<link rel="stylesheet" type="text/css"
href="catalog/view/theme/bt_claudine/stylesheet/bossthemes/bootstrap-
select.css" />
<!-- <link rel="stylesheet"
href="/catalog/view/theme/bt_claudine/stylesheet/credit.css"> -->
<link rel="stylesheet" type="text/css"
href="catalog/view/javascript/bossthemes/video/video.css">
<link
href="catalog/view/theme/bt_claudine/stylesheet/bossthemes/boss_category.css"
type="text/css" rel="stylesheet" media="screen" />
<link href="catalog/view/javascript/blog/colorbox/css/0/colorbox.css"
type="text/css" rel="stylesheet" media="screen" />
<link href="image/seocms.css" type="text/css" rel="stylesheet" media="screen"
/>
<link
href="catalog/view/theme/bt_claudine/stylesheet/bossthemes/boss_revolutionsli
der/css/settings.css" type="text/css" rel="stylesheet" media="screen" />
<link href="catalog/view/javascript/promotionlabelpro/style.css" type="text/css"
rel="stylesheet" media="screen" />
<link href="catalog/view/theme/bt_claudine/stylesheet/bossthemes/sly.css"
type="text/css" rel="stylesheet" media="screen" />
<link href="catalog/view/javascript/jquery/owl-carousel/owl.carousel.css"
type="text/css" rel="stylesheet" media="screen" />
<link
href="catalog/view/theme/bt_claudine/stylesheet/bossthemes/boss_megamenu.c
ss" type="text/css" rel="stylesheet" media="screen" />
<!-- <script type="text/javascript"
src="catalog/view/javascript/common.js"></script> -->
<script type="text/javascript"
src="catalog/view/javascript/bossthemes/cs.bossthemes.js"></script>
<script src="catalog/view/javascript/blog/colorbox/jquery.colorbox.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/blog/blog.color.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/bossthemes/sly/plugins.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/bossthemes/jquery.tools.min.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/bossthemes/jquery.revolution.min.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/bossthemes/sly/sly.js"
type="text/javascript"></script>

```



```

<script src="catalog/view/javascript/bossthemes/carouFredSel-6.2.1.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/jquery/owl-carousel/owl.carousel.min.js"
type="text/javascript"></script>
<script src="catalog/view/javascript/bossthemes/jquery.appear.js"></script>
type="text/javascript"

<script src="catalog/view/javascript/bossthemes/parallax/jquery.parallax-
1.1.3.js"></script>
<!-- You can add Google Analytics here or use T3 Injection feature -->
<script>

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
(i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new
Date();a=s.createElement(o),

m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
})(window,document,'script','https://www.google-
analytics.com/analytics.js','ga');

ga('create', 'UA-77087012-1', 'auto');
ga('require', 'displayfeatures');
ga('send', 'pageview');

</script>
<!-- Подключаем CallCons -->
<!-- script type="text/javascript"
src="https://prof.ua/callcons/js/callcons.js" charset='utf-8'></script -->
<script type="text/javascript"><!--
$(document).ready(function() {
$('#bt_loading').css("display", "none");
});
//--></script>

<style>
#bt_loading{position:fixed; width:100%; height:100%; z-index:999;
background:#fff}
.bt-loading{
height: 29px;
left: 50%;
margin-left: -120px;
margin-top: -29px;
position: absolute;

```

```

        top: 50%;
        width: 240px;
        z-index: 9999;
    }
</style>

<style type="text/css">
    body {
        background-color:#ffffff
    }
    a, body {
        color:#1c1c1c
    }
    a:hover, a:focus, .breadcrumb > li:hover a, .popular-cate .box-content
.sub_cat .sub_item a:hover {
        color:#dd5e65
    }
    .boss_homefilter_tabs .bt-heading h3, .open-panel i, .button-search:hover
    {
        color:#122A4D
    }
    .sub_image img, .btn-quickshop .sft_quickshop_icon, .boss-blog-
featured .image .time-stamp, .boss_homefilter_tabs .bt-heading h3:before,
#language:hover, #currency:hover, #cart.dropdown_cart:hover, .bt-account .box-
content, .bt-box .box-heading span:before, .htabs ul li:before, table.table thead
td, .wishlist-info thead td, .cart-info thead td, #collapse-checkout-confirm td
thead td, .article-layout .article-image .time-image, .title_heading:before,
div.jGrowl div.jGrowl-notification div.jGrowl-header, .b-
cart.dropdown_cart:before, .b-cart.dropdown_cart:hover, #back_top:hover span,
.language:hover, .currency:hover, .checkout .panel-default > .panel-heading,
.register fieldset h2:before, #article_related .article-image .time-image,
.menu_mobile,.b_bulk_title,.b_bulk_order_ajax table td .cart .btn {
        background-color:#122A4D
    }
    .b_bulk_title,.b_bulk_order_ajax table td .cart .btn,#boss_bulk_order
span.ui-corner-all {
        border-color:#404b8f
    }
    .btn, .btn-shopping, .btn-wishlist, .btn-compare, .button_cart:hover {
        background-color:#ffffff
    }
    .btn, .btn-shopping, .btn-wishlist, .btn-compare, .boss-blog-featured
.read_more a, .button_cart:hover {
        border-color:#bcbcbc
    }
    .btn, .btn-shopping, .btn-wishlist, .btn-compare, .boss-blog-featured
.read_more a, .button_cart:hover {
        color:#1c1c1c
    }

```

```

        .btn:hover, .btn-shopping:hover, .btn-wishlist:hover, .btn-compare:hover,
        ul.box-category > li.child span.plus, ul.box-category > li span.star, .bt-box .box-
        content .boss-alphabet ul li a:hover, .boss-blog-featured .read_more a:hover,
        .button_cart, .tp-caption.medium_text a.btn:hover, .btn-
        cart:hover,.b_bulk_order_ajax table td .cart .btn:hover {
            background-color:#dd5f65
        }
        .btn:hover, .btn-shopping:hover, .btn-wishlist:hover, .btn-compare:hover,
        .button_cart, .tp-caption.medium_text a.btn:hover, .btn-
        cart:hover,.b_bulk_order_ajax table td .cart .btn:hover {
            border-color:#dd5f65
        }
        .btn:hover, .btn-shopping:hover, .btn-wishlist:hover, .btn-compare:hover,
        .boss-blog-featured .read_more a:hover, .button_cart, .tp-caption.medium_text
        a.btn:hover, .btn-cart:hover,.b_bulk_order_ajax table td .cart
        .btn,.b_bulk_order_ajax table td .cart .btn:hover {
            color:#ffffff
        }
        .top_left, #top-links ul li a, #top-links ul li span, #language strong,
        #currency strong, #cart .btn-dropdown-cart b {
            color:#0a3b66
        }
        .top_left a, #top-links ul li a:hover, #top-links ul li a:hover span,
        .product-layout .article-name a:hover, .top-links ul li a:hover, .top-links ul li
        a:hover span {
            color:#122a4d
        }
        .mega-menu > ul.nav > li .dropdown .menu-row ul li:hover > .sub_menu,
        .boss_block-recent-post .time-stamp, .mega-menu > ul.nav > li .dropdown {
            border-color:#122a4d
        }
        .mega-menu > ul.nav > li > a, .mega-menu > ul.nav > li.parent > a i {
            color:#0a3b66
        }
        .mega-menu > ul.nav > li:hover > a, .mega-menu > ul.nav > li
        .dropdown .menu-column h3, .mega-menu > ul.nav > li .dropdown .menu-row
        a.parent {
            color:#122a4d
        }
        .mega-menu > ul.nav > li .dropdown .menu-row ul.column.category li a
        {
            color:#1c1c1c
        }
        .mega-menu > ul.nav > li .dropdown .menu-row ul.column.category >
        li:hover > a {
            color:#dd5f65
        }
        #bt_footer {
            background-color:#122A4D
        }
        footer h3 {
            color:#fff
        }
    
```

```

        footer ul li a, .footer-newsletter, .footer-newsletter .newsletter-content .btn,
        .powered-payment .nav_footer li a, #powered, footer a {
            color:#b4b4b4
        }
        footer a:hover, footer a:focus, .footer-newsletter .newsletter-content
        .btn:hover, .powered-payment .nav_footer li a:hover {
            color:#dd5f65
        }
        .name a, .product-layout .article-name a {
            color:#1c1c1c
        }
        .name a:hover {
            color:#dd5f65
        }
        .price, .product-thumb .price {
            color:#1c1c1c
        }
    </style>

```

```
<style type="text/css">
```

```

        body {
            font-family: Tahoma;
            font-size: 14px;
        }
        .btn {
        }
        #top-links
        ul li a, #top-links ul li span, #language span, #currency span {
        }
        footer h3,
        .footer-newsletter h3 {
        }
        footer a,
        .footer_column .footer-social ul li a i, .about-store p {
        }
        .mega-
        menu > ul.nav > li > a {
        }
        .mega-
        menu > ul.nav > li .dropdown .menu-row a.parent {
        }
    </style>

```

```

.name a {
}

.price {
}

h1, h2,
.bt-featured-pro.bt-nprolarge-tabs .htabs a, .bt-featured-pro .title span, .bt-box
.box-heading, .bt-featured h3, h2.title_border {
}

```

```
</style>
```

```
<style>
```

```

::-webkit-scrollbar {width: 8px; height: 8px; border: 3px solid #eee; }
::-webkit-scrollbar-button:start:decrement,          ::-webkit-scrollbar-
button:end:increment {display: block; height: 10px; background: #eee}
::-webkit-scrollbar-track {background: #eee; -webkit-border-radius: 10px;
border-radius:10px; -webkit-box-shadow: inset 0 0 4px rgba(0,0,0,.2)}
::-webkit-scrollbar-thumb {height: 50px; width: 50px; background:
rgba(0,0,0,.5); -webkit-border-radius: 8px; border-radius: 8px; -webkit-box-
shadow: inset 0 0 4px rgba(0,0,0,.1);}

```

```
</style>
```

```
<link
```

```

href="catalog/view/javascript/bossthemes/fancybox/jquery.fancybox.css"
type="text/css" rel="stylesheet" media="screen" />

```

```
<script
```

```

src="catalog/view/javascript/bossthemes/fancybox/jquery.fancybox.js"
type="text/javascript"></script>

```

```
<!--
```

```
<script
```

```
type="text/javascript"
```

```

src="catalog/view/javascript/inneti.js"></script> -->

```

```
<script type="text/javascript">
```

```
$(document).ready(function() {
```

```
$("#inn_size").fancybox({
```

```
'titlePosition' : 'inside',
```

```
'transitionIn' : 'none',
```

```
'transitionOut' : 'none'
```

```
});
```

```
$('.dropdown_cart').click(function(){
```

```
$('.btn-dropdown-cart').click();
```

```
// console.log('=');
```

```
});
```

```
$('.low_price_cat').click(function () {
```

```

        // console.log(1111);
        $("#inn_sizeq").fancybox({
            'titlePosition' : 'inside',
            'transitionIn' : 'none',
            'transitionOut' : 'none'
        });
        var t = $(this).parent().parent().find('.name').text();
        var a = $(this).parent().parent().find('.artikul').text();
        // console.log(t);
        $('#input-enquiry').text('Назва: '+ t + "\n\r" + a);
    });
</script>

```

```

<script type="application/ld+json">
    {
        "@context": "https://schema.org",
        "@type": "LocalBusiness",
        "name": "PROF",
        "alternateName": "2IIP",
        "url": "://",
        "logo": "https://prof.ua/image/catalog/banner/PROF-logoNew-
min.png",
        "image": "https://prof.ua/image/catalog/banner/PROF-logoNew-
min.png",
        "@id": "",
        "telephone": "068-844-11-44",
        "contactPoint": [{
            "@type": "ContactPoint",
            "telephone": "073-844-11-44",
            "contactType": "customer service",
            "availableLanguage": "Ukrainian"
        },{
            "@type": "ContactPoint",
            "telephone": "068-844-11-44",
            "contactType": "sales",
            "availableLanguage": "Ukrainian"
        }],
    },

```

```

        "sameAs":
        "https://www.facebook.com/PROF.tools/?ref=aymt_homepage_panel&eid=AR
        AyF9rchk92l0rRup6uVcoqLyTtl6IKVnfm2jI-
        O1F4bup9NVPIm68CiHcBTq4yQZrKs7gz-кес_TB4",
        "address": {
            "@type": "PostalAddress",
            "streetAddress": "Манастирського, 2",
            "addressLocality": "Львів",
            "postalCode": "79066",
            "addressCountry": "UA"
        },
        "openingHoursSpecification": {
            "@type": "OpeningHoursSpecification",
            "dayOfWeek": [
                "Monday",
                "Tuesday",
                "Wednesday",
                "Thursday",
                "Friday",
                "Saturday"
            ],
            "opens": "08:30",
            "closes": "18:00"
        }
    }
</script>

```

```
<link rel="alternate" hreflang="uk" href="https://prof.ua/" />
```

```
<link rel="alternate" hreflang="ru" href="https://prof.ua/ru/" />
</head>
```

```

<body class="bt-home-page ">
  <!-- Start SiteHeart code -->
  <!-- <script>
  (function(){
    var widget_id = 839084;
    _shcp =[{ widget_id : widget_id}];
    var lang =(navigator.language || navigator.systemLanguage
    || navigator.userLanguage ||"en")
    .substr(0,2).toLowerCase();
    var url  ="widget.siteheart.com/widget/sh/" +  widget_id  + "/" +  lang
    + "/widget.js";
    var hcc = document.createElement("script");
    hcc.type ="text/javascript";

```

```

    hcc.async =true;
    hcc.src =("https:"== document.location.protocol ?"https":"http")
    +"://" + url;
    var s = document.getElementsByTagName("script")[0];
    s.parentNode.insertBefore(hcc, s.nextSibling);
  }());
</script> -->
<!-- End SiteHeart code -->
<div id="bt_container" class="bt-wide" >
<div id="bt_header" class="" >
<nav id="top">

<div class="container">

    <div class="b_mobile">
        <a class="open-panel"><i class="fa fa-bars"></i></a>
        <div class="menu_mobile">
            <a class="close-panel"><i class="fa fa-times-circle"></i></a>
                <div class="logged-link">
                    <a href="index.php?route=account/login"><i class="fa fa-
user"></i><span>Залогуватись</span></a>
                    <a
href="index.php?route=account/register"><i class="fa fa-
plus"></i><span>Створити аккаунт</span></a>
                </div>
                <div class="language_currency">
                    <div class="pull-left">

<div class="btn-group">
    <button class="btn btn-link dropdown-toggle" data-toggle="dropdown">
        
        <span class="hidden-xs hidden-sm hidden-md">Мова</span> <i
class="fa fa-caret-down"></i></button>
        <ul class="dropdown-menu">
            <li><a href="https://prof.ua/#"> УКР</a></li>
            <li><a href="https://prof.ua/ru/"> РУС</a></li>
        </ul>
    </div>
</div>
</div>

        <!-- <div id="search" class="input-group">
            <input type="text" name="search" value="" placeholder="Пошук"/>
            <button title="Search" type="button" class="button-search"><i class="fa
fa-search"></i></button>
        </div> -->

    </div>

```



```

<div class="top-links">
  <ul class="list-inline">
    <li class="contact"><a
href="https://prof.ua/index.php?route=information/contact">
      <i class="fa fa-phone"></i></a> <span
class="hidden-xs hidden-sm hidden-md">073-844-11-44<br>068-844-11-
44</span>
    </li>
    <li
class="dropdown"><a href="https://prof.ua/index.php?route=account/account"
title="Акаунт" class="dropdown-toggle" data-toggle="dropdown"><i class="fa
fa-user"></i><span>Акаунт</span><i class="fa fa-angle-down"></i></a>
      <ul class="dropdown-menu dropdown-menu-right">
        <li><a
href="https://prof.ua/index.php?route=account/register">Реєстрація</a></li>
        <li><a
href="https://prof.ua/index.php?route=account/login">Вхід</a></li>
      </ul>
    </li>

    <li>
      <a
href="https://prof.ua/index.php?route=account/wishlist" class="wishlist-total"
title="Список побажань (0)"><i class="fa fa-heart heart-wishlist" aria-
hidden="true"></i> <!-- <span>Список побажань (0)</span> --> </a></li>
    </ul>
  </div>

  <div class="bt_mainmenu">

    <nav class="mega-menu">
      <ul class="nav nav-pills">
        <li class="parent">
          <p class="plus visible-xs">+</p>
          <a
href="https://prof.ua/benzoinstrumenty"><span class="menu-title">Бензо<i
class="fa fa-sort-down"></i></span><span class="label label-success"
></span></a>
          <div class="dropdown drop-grid-6-2">

            <div
class="menu-row row-col-2" >

              <div class="menu-column row-grid-2">

```

```

<!-- category -->

<a href="https://prof.ua/benzoinstrumenty" class="parent">
    Бензоінструменти
</a>
<ul class="column category">

    <li class="col-grid-2 ">
        <a
href="https://prof.ua/benzopulu">
            Бензопили
            <i class="fa fa-caret-
right"></i></a>
        </li>

    <li class="col-grid-2 ">
        <a
href="https://prof.ua/benzorisys">
            Бензорізи
            <i class="fa fa-caret-
right"></i></a>
        </li>

    <li class="col-grid-2 ">
        <a
href="https://prof.ua/dvyhuny">
            Двигуни
            <i class="fa fa-caret-
right"></i></a>
        </li>

    <li class="col-grid-2 sub_category">
        <a
href="https://prof.ua/komplektuuychi_vytratni">
            Комплектуючі та
            витратні матеріали для бензоінструментів
            <i class="fa fa-caret-right"></i></a>
        <div
class="sub_menu"><ul><li
href="https://prof.ua/komplektuuychi_vytratni/lantsyuhy_do_pyl">Ланцюги до
пил</a></li><li
href="https://prof.ua/komplektuuychi_vytratni/napylnyky">Напильники</a></l
i><li ><a href="https://prof.ua/komplektuuychi_vytratni/shyyny_do_pyl">Шини

```

```

до пил</a></li></ul></div>
  </li>
    <li class="col-grid-2 ">
      <a
href="https://prof.ua/kyltuvatoru">
        Культиватори
        <i
class="fa fa-caret-right"></i></a>
      </li>
    <li class="col-grid-2 ">
      <a
href="https://prof.ua/kystorizu">
        Кущорізи бензинові
        <i class="fa fa-caret-
right"></i></a>
      </li>
    <li class="col-grid-2 ">
      <a
href="https://prof.ua/mastyla">
        <i
class="fa fa-caret-
right"></i></a>
      </li>
    <li class="col-grid-2 ">
      <a
href="https://prof.ua/motobloky">
        <i
class="fa fa-caret-
right"></i></a>
      </li>
    <li class="col-grid-2 ">
      <a
href="https://prof.ua/motobury-benzoinstrumenty">
        <i
class="fa fa-caret-
right"></i></a>
      </li>
    <li class="col-grid-2 ">
      <a
href="https://prof.ua/motopompy">

```

```

right"></i></a>
</li>
<li class="col-grid-2 ">
href="https://prof.ua/obpruskyvachi">
бензинові
class="fa fa-caret-right"></i></a>
</li>
<li class="col-grid-2 ">
href="https://prof.ua/benzotrymery">
бензинові
class="fa fa-caret-right"></i></a>
</li>
<li class="col-grid-2 ">
href="https://prof.ua/hazonokosarky">
бензинові
class="fa fa-caret-right"></i></a>
</li>
</ul>
</div>
</div>
</div>
</li>
<li class="parent">
<p class="plus visible-xs">+</p>
href="elektroinstrumenty"><span class="menu-title">Електро<i class="fa fa-
sort-down"></i></span><span class="label label-success" ></span></a>
<div class="dropdown drop-grid-6-2">
<div
class="menu-row row-col-2" >
<div class="menu-column row-grid-2">
<!-- html-->
<div class="staticblock"><ul
class="column category">

```

Мотопомпи

<i class="fa fa-caret-

</li>

<li class="col-grid-2 ">

href="https://prof.ua/obpruskyvachi">

бензинові

class="fa fa-caret-right"></i></a>

</li>

<li class="col-grid-2 ">

href="https://prof.ua/benzotrymery">

бензинові
class="fa fa-caret-right"></i></a>

</li>

<li class="col-grid-2 ">

href="https://prof.ua/hazonokosarky">

бензинові

class="fa fa-caret-right"></i></a>

</li>

</ul>

</div>

</div>

</div>

</li>

<li class="parent">

<p class="plus visible-xs">+</p>

<a

href="elektroinstrumenty"><span class="menu-title">Електро<i class="fa fa-
sort-down"></i></span><span class="label label-success" ></span></a>

<div class="dropdown drop-grid-6-2">

<div

class="menu-row row-col-2" >

<div class="menu-column row-grid-2">

<!-- html-->

<div

class="staticblock"><ul

class="column category">