

Міністерство освіти і науки України

Луцький національний технічний університет

(повне найменування закладу вищої освіти)

Факультет комп'ютерних та інформаційних технологій

(повне найменування факультету)

Кафедра комп'ютерної інженерії та кібербезпеки

(повне найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**WEB-SERVICE МАГАЗИНУ ЗАСОБАМИ JAVASCRIPT ДЛЯ
ОРГАНІЗАЦІЇ ТОРГІВЛІ НА ЗАМОВЛЕННЯ**

**WEB-SERVICE OF THE STORE BY MEANS JAVASCRIPT THE
FOR ORGANIZATION OF TRADE TO ORDER**

спеціальність 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія
(назва освітньої програми)

Виконав: здобувач вищої освіти
групи КІ-41
Сорока Юрій Олександрович

(підпис)

Керівник:
к.т.н., доцент
Пех Петро Антонович

(підпис)

Кваліфікаційну роботу
допущено до захисту
« _____ » червня _____ 2023 р.
Гарант освітньої програми:
к.т.н., доцент
Лавренчук Світлана Василівна

(підпис)

Луцьк – 2023 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра комп'ютерної інженерії та кібербезпеки

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 123 Комп'ютерна інженерія

Освітня програма: «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ проф. Н.Черняшук

« _____ » _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Сороці Юрію Олександровичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи _____ *Web-сервіс магазину засобами JavaScript для організації торгівлі на замовлення*

Керівник роботи _____ *Пех Петро Антонович, к.т.н., доцент*

затверджені наказом закладу вищої освіти від «28» грудня 2022 року № 982/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи _____ 01.06.2023р.

3. Вихідні дані до роботи _____ *Література в області захисту даних та поведінкового аналізу зарубіжні та вітчизняні роботи в обраній сфері розробки*

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

Вступ

Аналіз задачі створення веб-сайту

Аналіз та вибір засобів розробки

Розробка веб-сайту для торгівлі на замовлення

Висновки

5. Перелік графічного (ілюстративного) матеріалу:

Знімки екрану інтерфейсу відвідувача веб-сайту та інструментів розробки.

Лістинг коду

Таблиці графічних характеристик використання

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Загальний аналіз створення веб-сайту</i>	<i>Пех П.А.</i>		
<i>Аналіз та вибір засобів розробки</i>	<i>Пех П.А.</i>		
<i>Розробка веб-сайту для торгівлі на замовлення</i>	<i>Пех П.А.</i>		
<i>Висновки</i>			

7. Дата видачі завдання 01.11.2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	<i>Обґрунтування теми</i>	13.11.2022 р.	Виконано
2.	<i>Огляд літератури на тему розробки телеграм ботів</i>	17.11.2022 р.	Виконано
3.	<i>Дослідження предметної області</i>	29.11.2022 р.	Виконано
4.	<i>Вибір засобів розробки</i>	11.12.2022 р.	Виконано
5.	<i>Розробка та тестування веб-сайту</i>	24.01.2023 р.	Виконано
6.	<i>Оформлення матеріалів роботи</i>	29.03.2023 р.	Виконано
7.	<i>Нормоконтроль</i>	18.05.2023 р.	Виконано
8.	<i>Інструментальна перевірка на академічний плагіат</i>	01.6.2023 р.	Виконано
9.	<i>Представлення кваліфікаційної роботи бакалавра до захисту</i>	07.06.2023 р.	Виконано

Здобувач вищої освіти

(підпис)

Сорока Ю.О.

(прізвище, ініціали)

Керівник кваліфікаційної роботи

(підпис)

Пех П.А.

(прізвище, ініціали)

АНОТАЦІЯ

Сорока Ю.О. Web-сервіс магазину засобами JavaScript для організації торгівлі на замовлення.

Кваліфікаційна робота бакалавра ОП «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія. Луцький національний технічний університет. Луцьк, 2023.

Кваліфікаційна робота складається з вступу, трьох розділів, висновків, списку використаних джерел та двох додатків.

У розділі 1 були надані загальні відомості про предметну область та виконано порівняльний аналіз аналогічних веб-сайтів. Аналізуючи ці сайти, було виявлено їх переваги та недоліки.

У 2 розділі було проведено детальний аналіз інструментів та програм розробки. Було обґрунтовано вибір технологій для реалізації веб-сайта, середовище розробки, а також бібліотеки які були використані у ході роботи.

У розділі 3 було проведено безпосередню розробку веб-сайту. Було створено сайт, поставлено технічні вимоги для його використання, а також розглянуто принцип його роботи .

Об'єкт дослідження – методи та засоби розробки веб-сайту для торгівлі.

Предмет дослідження – веб-сайт магазину засобами JavaScript для організації торгівлі на замовлення.

Метою роботи є модернізація та удосконалення сучасних методів розробки веб-сайту інтернет-магазину із продажу товарів на замовлення і на базі отриманої інформації створити власний веб-сайт.

Ключові слова: веб-сайт, веб-сторінка, JavaScript, головна сторінка, бібліотека, HTML, CSS.

ANNOTATION

Soroka Y.O. Web-service of the store by means of JavaScript for the organization of trade to order.

Bachelor's qualifying thesis of the OP "Computer Engineering" specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2023.

The qualification work consists of an introduction, three sections, conclusions, a list of used sources, and three appendices.

Section 1 provided an overview of the subject area and a comparative analysis of similar websites. By analyzing these sites, their advantages and disadvantages were revealed.

In section 2, a detailed analysis of development tools and programs was carried out. The choice of technologies for the implementation of the website, the development environment, as well as the libraries that were used during the work were justified.

Section 3 involved the direct development of the website. The site itself was created, technical requirements for its use were set, and the principle of its operation was also considered. The object of research is methods and means of developing a website for trade. The subject of the study is a website of a store using JavaScript for the organization of trade to order.

The purpose of the work is to modernize and improve modern methods of developing a website for an online store selling goods to order and to create your own website based on the information received.

Keywords: website, web page, JavaScript, homepage, library, HTML, CSS.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ЗАДАЧІ СТВОРЕННЯ ВЕБ-САЙТУ.....	9
1.1 Поняття веб-сайту.....	9
1.2 Сайт як маркетинговий інструмент.....	11
1.3 Дизайн веб сайту.....	14
1.4 Зручність використання веб-сайту.....	15
1.5 Важливість веб-сайту.....	17
1.6 Аналіз існуючих аналогів.....	18
1.6.1 Веб-сайт піцерії «Теревені».....	18
1.6.2 Веб-сайт піцерії «Pronto Pizza».....	19
РОЗДІЛ 2 АНАЛІЗ ТА ВИБІР ЗАСОБІВ РОЗРОБКИ.....	21
2.1 Вибір середовища та інструментів розробки.....	21
2.1.1 Огляд інтервейсу VS Code.....	21
2.1.2 HTML.....	22
2.1.3 CSS.....	25
2.1.4 LESS.....	26
2.1.5 JavaScript.....	27
2.1.6 AJAX.....	29
2.1.7 JQuery.....	29
2.1.8 BootStrap.....	30
2.2 Системи управління веб-контентом.....	31
2.2.1 WordPress.....	31
2.2.2 Joomla.....	32
2.2.3 Drupal.....	32
РОЗДІЛ 3 РОЗРОБКА ВЕБСАЙТУ ДЛЯ ТОРГІВЛІ НА ЗАМОВЛЕННЯ.....	33
3.1 Огляд головної сторінки веб-сайту.....	33
3.1.1 Заголовок веб-сайту.....	33
3.1.2 Тіло веб-сторінки.....	35

3.1.3 Footer веб-сайту.....	36
3.2 Навігація на веб-сайті.....	38
3.3 Технічні вимоги до середовищ використання.....	39
3.4 Принцип роботи веб-сайту.....	39
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	41
ДОДАТКИ.....	43
Додаток А Лістинг коду файлу index.html.....	44
Додаток Б Лістинг коду файлу style.less.....	50

ВСТУП

Актуальність теми. Технології не стоять на місці, все більше стає поширеною купівля будь-яких товарів в інтернет-магазинах, багато компаній прагнуть перевести свій бізнес в онлайн, але серед великої кількості веб-ресурсів знайти дійсно зручну платформу стає все більш важче. Дана кваліфікаційна робота є затребуваною, оскільки вона вносить свій вклад у розвиток технологій розробки веб-сайтів. Досліджуючи технології розроблення веб-сайтів для продажу товарів на замовлення, ця робота має на меті саме їх модернізацію та удосконалення.

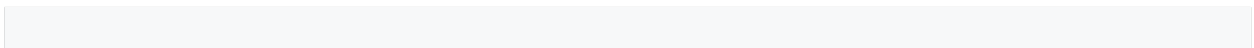
Метою роботи є модернізація та удосконалення сучасних методів розробки веб-сайту інтернет-магазину із продажу товарів на замовлення і на базі отриманої інформації створити власний веб-сайт.

Об'єкт дослідження – методи та засоби розробки веб-сайту для торгівлі.

Предмет дослідження – вивчення особливостей створення веб-сайтів за допомогою мов програмування HTML, CSS та JavaScript, для продажі товарів на замовлення.

Були сформовані наступні завдання на кваліфікаційну роботу:

- провести аналіз існуючих аналогових веб-сайтів;
- провести аналіз етапів веб-розробки;
- здійснити аналіз існуючих інструментів для створення веб-сайту;
- розробити дизайн та програмну реалізацію веб-сайту;



РОЗДІЛ 1

АНАЛІЗ ЗАДАЧІ СТВОРЕННЯ ВЕБ-САЙТУ

1.1 Поняття веб-сайту

Веб-сайт – це сукупність різних веб-сторінок, які об'єднанні за змістом та навігацією і мають одне доменне ім'я, наприклад, www.youtube.com. Веб-сайт і веб-сторінка використовуються взаємозамінно, але один веб-сайт може містити безліч веб-сторінок.

Для прямого доступу клієнтів до веб-сайтів на серверах був спеціально розроблений протокол HTTP. Крім веб-сайтів в мережі Інтернет так само доступні WAP-сайти для мобільних телефонів.

Всі веб-сайти Інтернету в сукупності складають Всесвітню павутину. Веб-сайт є колекцією документів, графіків чи інформації, яку користувачі можуть переглянути до їх потреб.

«Здебільшого, одному сайту відповідає одне доменне ім'я. Саме завдяки доменним іменам, ми можемо знайти потрібний нам сайт в Всесвітній павутині. Як було сказано вище, переважно всі сайти мають одне доменне ім'я, бувають і інші варіанти, такі як: один сайт на декількох доменах, або декілька сайтів на одному домені. Декілька доменів, в більшості, використовують великі сайти, щоб зрозуміло відокремити різні види послуг. Такими сайтами є: mail.google.com, maps.google.com, а також багато інших великих сайтів» [1]. «Буває й таке, що один іменний домен може об'єднувати декілька сайтів, характерно це для безкоштовних хостингів» [2].

«Всі веб-сайти зберігаються на веб-серверах. Веб-сервер – частина Всесвітньої павутини. Веб-сервера приймають HTTP-запити від юзерів, переважно це відбувається в веб-браузері, та повертає їм HTTP-відповідь з HTML сторінкою» [3].

Іноді, сайт може бути доступний за різними адресами і зберігатись на різних серверах. Дублікат такого сайту називається дзеркалом. Існують

також офлайн версії сайту. Такі сайти можна відкрити на своєму пристрої без підключення до мережі і використання серверного ПЗ.

У 1989 році британський інженер-програміст, який працював підрядником CERN(Європейська організація ядерних досліджень), розробив революційну ідею для вчених, щоб вони могли вільно обмінюватися документами та текстами через географічні кордони. Ця ідея дозволила отримати інформаційний обмін між комп'ютерами за допомогою інтернету та нової технології – гіпертекст. Спочатку ця ідея була розроблена спеціально для академіків і вчених, щоб вони мали змогу ділитися академічними документами та файлами один з одним, але ця ідея швидко розвинулась у всесвітню мережу. Це змінило спосіб нашого спілкування, спосіб ведення бізнесу, спосіб здійснення покупок, отже, www здійснила революцію в житті мільярдів людей, так як його творець, Тім Бернерс Лі, не міг цього передбачити. Спочатку www було запропоновано в 1989 році, і було запущено всього через два роки, на початку 1991. Перша веб-сторінка була випущена того ж року, її дизайн (рис 1.1) був дуже простий. На цій сторінці відображалася інформація, щодо проекту www, пояснення, як працює www, хто брав участь у розвитку проекту, та призначення гіпертексту. Сторінка була дуже простою, на ній не було зображень, спадних меню чи наведення миші, на ньому не було елементарної естетики дизайну, яка, сьогодні, сприймається як належне.

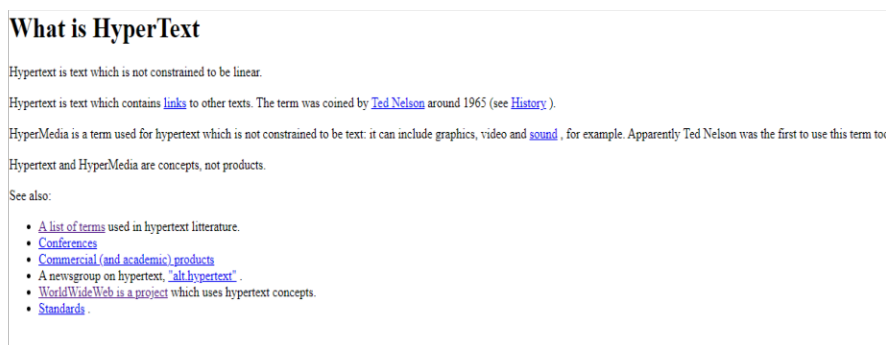


Рисунок 1.1 – Вигляд першого веб-сайту

З перших днів веб-розробка пройшла довгий шлях. Веб-розробка стала окремою дисципліною, оскільки все більше людей підключаються до інтернету і WWW, кількість веб-сторінок і онлайн-контенту зростає в геометричній прогресії. «Наразі, WWW є доменом для понад 170 мільйонів активних веб-сайтів, і ця кількість продовжує зростати з кожним днем.» [4]

Разом із цим зростанням, ми спостерігаємо розвиток галузі веб-розробки. У веб-розробці також було досягнуто важливих успіхів. Було розроблено нові мови, такі як JavaScript, які дозволяють розробникам створювати красиві динамічні веб-сторінки. Крім того, розробляються бібліотеки, щоб зробити кодування більш інтуїтивним, зрозумілим, легшим для написання, а веб-сторінки швидшими. Крім цих мов і бібліотек, інструменти розробника, такі як IDE (інтегроване середовище розробника), допомагають зручно створювати проекти. Усі ці зміни допомогли розробникам підвищити ефективність і розширили їхню здатність створювати красиві, професійно виглядаючі та оригінальні веб-сайти.

Існує два типи веб-сайтів: статичні та динамічні. Статичні сайти, зазвичай, створюються за допомогою таких мов програмування, як HTML та CSS. Статичні веб-сайти сьогодні вкрай рідкісні. Тоді як динамічні сайти стали популярними завдяки зручності використання та функціональності, яка легко і швидко оновлюється. Динамічний веб-сайт легко підтримувати через простий інтерфейс браузера.

1.2 Сайт як маркетинговий інструмент

Веб-сайт – це сукупність різних веб-сторінок, які працюють під одним доменом назву, наприклад google.com. Веб-сайт і веб-сторінки використовуються взаємозамінно, але одне веб-сайт може містити багато веб-сторінок. Веб-сайт є колекцією документів, графіків чи інформації, яку може отримати користувач.

Наявність веб-сайту стало необхідним для будь-якого бізнесу, оскільки Інтернет доступний до кожного. Згідно з Котабле та Хельсеном, використання ІКТ у маркетингу розвивається практикувався в електронному бізнесі та електронній комерції в 1990-х роках. Електронна комерція через Інтернет має змінити очікування клієнтів щодо ціни, порівнянності, швидкості та послуг. Інтеграція ІКТ з бізнесом підвищує ефективність і конкурентоспроможність компанії. Він може зв'язувати клієнтів, співробітників і постачальників за допомогою комунікацій та угода. ІКТ відіграють значну роль у маркетингу для створення цінності клієнтів.

У наш час маркетинг є важливим аспектом будь-якого бізнесу, для досягнення поставленої успішно мети. Існують інші способи маркетингу для бізнесу, але з моменту їх наявності, в інтернеті стало все більш поширеним використання веб-сайту одним із найпотужніших маркування інструменти для бізнесу. Інтернет і веб-сайт відіграють життєво важливу роль у веденні бізнесу простіше і доступніше, ніж раніше. Створення власного веб-сайту компанії може бути кращим способом для компанії у спілкуванні зі своїми потенційними клієнтами. Веб-сайт відіграє важливу роль у відносинах між клієнтами та компаніями, шляхом спільного використання та обміну інформацією про продукт або послуги.

Веб-сайт є важливим джерелом інформації про компанію. Його можна інтегрувати з мультимедійними функціями, такими як текст, документи, аудіо та відео як інформаційне джерело. Це допомагає компанії розширити існуючий бізнес, просуваючи свою продукцію та обслуговування багатьом клієнтам. Оскільки веб-сайт доступний у будь-який час і з в будь-якому місці, компанія зможе взаємодіяти з клієнтами, розуміти їхні потреби і негайно отримувати відповіді.

Компанії створюють веб-сайти для різних цілей. В основному є чотири типи онлайн веб-сайти присутності. Кожен вид веб-сайту створений для різних цілей у викоримтанні. Веб-сайт, який дозволяє клієнтам купувати продукти в інтернеті, називають транзакційним веб-сайтом електронної

комерції. Ebay.com або amazon.com є чудовим прикладом транзакційного веб-сайту електронної комерції. Клієнт може переглядати продукти, і одразу можуть зробити покупку за допомогою наданих методів онлайн-платежів. Сервісно-орієнтована компанія будує їхній веб-сайт, де клієнт може не мати змоги придбати продукти, але він може отримати інформацію про бажану послугу через сайт. Ці типи веб-сайтів мають назву «Веб-сайт, орієнтований на обслуговування». Третій тип веб-сайтів, сервіси для створення бренду, де клієнт може побачити продукт онлайн, але не може зробити будь-яку покупку. Результатом цього дипломного проекту є створення веб-сайту для створення бренду де на сайті буде доступна вся інформація про товар, але клієнти не зможуть здійснити покупку. Покупці можуть знайти місце розташування магазину через веб-сайт і відвідайте магазин самостійно, щоб зробити покупку. Є ще один тип веб-сайту, який називається порталом або медіасайтом. Цей тип веб-сайту надає змогу клієнтові переглянути інформацію, яка йому потрібна.

Компанії, які обирають веб-сайт як інструмент маркетингу, мають можливість спілкуватися зі своїми потенційними клієнтами по всьому світу. Ось чому маркетинг через веб-сайти стають все більш популярними порівняно з традиційним способом маркетингу, наприклад брошури, плакати, реклама в газетах чи радіотелебаченнях, яке коштує набагато торожче, ніж інтернет магазин. Клієнт може знайти на вебсайт за допомогою інтернет браузерів, наприклад: google, mozilla, opera, тощо. Це є набагато зручніше, аніж простий маркетинг. Веб-сайт також є способом маркетингу постійний. Одного разу зроблене воно триває. На відміну від будь-якого паперового маркетингу, як брошури, газета реклами, плакатів, а також телевізійної та радіореклами, її не потрібно переробляти або повторюється. «Веб-сайт також є найкращим способом розвинути бізнес у всьому світі. Через Інтернет, легко зв'язатися з людьми по всьому світу» [5].

1.3 Дизайн веб сайту

Розробка веб-сайту так само важлива, як і зміст та інформація яка на ньому знаходиться. Веб-сайт має бути створений таким чином, щоб клієнти могли знаходити потрібну їм інформацію без будь-яких відволікань. Дизайн і зміст повинні відображати мету аудиторія та мета сайту. Кожен клієнт, який відвідує веб-сайт, має різні навички, здібності та думки. Ось чому правильний дизайн веб-сайту має важливе значення для найкращої взаємодії з користувачем. «Якщо веб-сайт не відповідає потребам призначених користувачів, це не відповідатиме потребам організації, що надає веб-сайт. Розробка веб-сайту повинна бути зосереджена на користувачеві, оцінюючи розвиток дизайну щодо користувача вимоги» [6]. Цей веб-дизайн, орієнтований на користувача, пропонував відповіді на кілька запитання перед створенням дизайну сайту. Питання наступні:

- Хто є основними користувачами?
- Яка мета доступу до сайту?
- Як часто вони будуть відвідувати сайт?
- Який досвід і знання вони мають?
- Якої вони національності?
- Вони вміють читати українською?
- Яку інформацію вони шукають?
- Як вони захочуть використати інформацію: прочитати її на екрані, роздрукувати або завантажити?
- Які типи браузерів вони використовуватимуть?
- Наскільки швидкими будуть їхні зв'язки?
- Який великий екран/вікно вони використовуватимуть, зі скількома кольорами?

Відповіді на ці запитання допоможуть розробнику створити необхідний дизайн а веб-сайт, який буде служити інтересам компанії або користувачів. Однак там це дві основні точки зору, з яких більшість людей визначають, чи

є дизайн веб-сайту «Добре чи погано». Один – це зручність використання, яка зосереджена на функціональності, ефективності, правильності подання інформації, оперативність. Інший – це естетична перспектива, і це все про презентацію, анімацію та графіку. Є веб-сайти, які переповнені анімації та графіки, але вони не вважаються добре розробленим веб-сайтом, оскільки функціональності та ефективності. Через це відвідувачі веб-сайту можуть загубитися або заплутатися занадто багато анімації та графіки. Найважливішою частиною розробки веб-сайту є спілкування з відвідувачами. Ось чому веб-сайт розроблено з урахуванням зручності використання точки зору вважають хорошим дизайном веб-сайту.

1.4 Зручність використання веб-сайту

Простота використання – це концепція, яка встановлює свою позицію щодо зв'язку між кутом зору людини до інформаційно-комунікаційні технології. Зручність використання також можна розглядати як дизайн підхід, атрибут продукту та вимірювання. На деяких веб-сайти користувачам важко орієнтуватися, і через це важко знайти інформацію складний дизайн. Ці неякісні веб-сайти витрачатимуть час користувачів і можуть їх розчарувати. Неякісний веб-сайт може справити негативне враження на цілій компанії. Ось чому зручність використання є важливою для веб-сайту.

Мета компанії щодо створення веб-сайту не буде виконана, якщо веб-сайт не відповідає потребам користувачів.

Найкращим і поширеним, ймовірно, є визначення Якоба Нільсена, який описує зручність використання як атрибут якості, який оцінює, наскільки легкий користувача інтерфейс сайту. Для досягнення конкретної мети компанії, деякі характеристики простоти використання, які слід дотримуватися, щоб зробити сайт зручним для користувача. На рисунку 1.2 показано графічне представлення характеристик зручності використання:

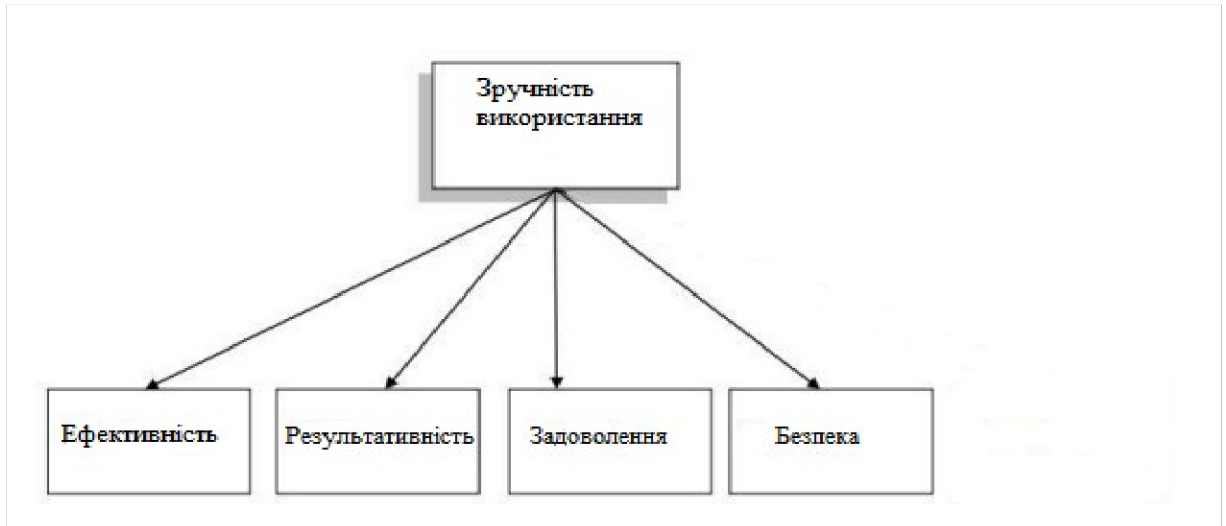


Рисунок 1.2 — Графічний представник характеристик зручності використання

Характеристики зручності використання веб-сайту описані нижче:

– Ефективність: користувач повинен мати можливість використовувати веб-сайт без будь-яких труднощів оскільки пересічний користувач Інтернету не розуміє технічних аспектів на веб-сайті, і вони можуть швидко втратити інтерес, якщо заплутаються в цій інформації на одній із сторінок. Дизайнер повинен використовувати простий і зрозумілий інтерфейс, який буде більш корисним для всіх типів користувачів або клієнтів, які використовують цей веб-сайт.

– Результативність: результативність веб-сайту можна виміряти затracеним часом і точністю для виконання завдання, запропонованого користувачем. Результативність є фундаментальним питанням у будь-якому дизайн веб-сайту, якщо веб-сайт працює повільно або занадто довго відкривається тоді користувач може втратити терпіння залишатися на тому самому веб-сайті. Це може статися, якщо веб-сайт містить велику кількість анімації або графіки, що призводить до повільного завантаження інформації сторінці. Швидкість Інтернету не в усьому світі однакова. У районі, де швидкість Інтернету низька, їм буде важко переглядати веб-сайт, який містить величезну кількість зображень або анімації високої чіткості.

– Задоволення: користувач має отримувати задоволення від використання веб-сайту. Як приклад: клієнт відвідує веб-сайт, і хоче отримати інформацію про продукт, який його цікавить, веб-сайт повинен мати всю інформацію про цей продукт, щоб задовольнити такого клієнта. Якщо ж цієї інформації не буде на сайті, то клієнт може втратити бажання придбати цей товар. Веб-сайтом вважається задовільно використовувати, коли користувач не відчуває ніякого дискомфорту чи негативного ставлення щодо використання веб-сайту.

– Безпека: користувач повинен відчувати себе в безпеці, коли відвідує веб-сайт. При виготовленні веб-сайт для електронної комерції, усі способи оплати мають бути автентифіковані та користувач повинен почуватися безпечно, користуючись цими способами оплати. Інтернет стає все менш безпечним. Веб-дизайнер повинен сумлінно ставитися до розміщення шкідливих посилань на веб-сайт.

1.5 Важливість веб-сайту

У сучасну епоху люди настільки прогресивні, що завжди шукають хороші речі чи продукти. Всі хочуть отримувати кращі речі, ніж інші. Щоразу, коли люди йдуть до ресторану, щоб поїсти, вони завжди вибирають ту їжу, яка їм найбільше сподобалась на картинці. Точно те ж саме відбувається, коли люди заходять на сайт. Більшість людей купляють собі речі в таких інтернет-магазинах, як Amazon, e-bay, Apple, LinkedIn, адже вони є популярними і надійними. Насправді, наявність веб-сайту малої чи великої компанії означає, що вона має власну ідентифікацію, щоб продемонструвати іншим свою продукцію.

1.6 Аналіз існуючих аналогів

На сьогоднішній день, у Всесвітній павутині є безліч сайтів, які спеціалізуються на продажі товарів через веб-сайти, у цьому розділі, моєю головною метою є аналіз переваг та недоліків систем готових веб-сервісів, які допоможуть мені реалізувати власний веб-сайт.

1.6.1 Веб-сайт піцерії «Теревені»

Розглядаючи подібні інтернет магазини, найбільше мені сподобався сайт піцерії «Теревені», адже він є відносно простий у розробці, і має просту навігацію. На головній сторінці (рис. 1.3), знаходиться інформація про меню, доставку, контактні дані, та акції які наразі діють. Для того, щоб переглянути асортимент, потрібно натиснути на вкладку «Меню» (рис 1.4), після чого, обрати потрібний розділ. На сайті немає авторизації, тож, щоб замовити потрібний товар, потрібно дзвонити їм напряму.

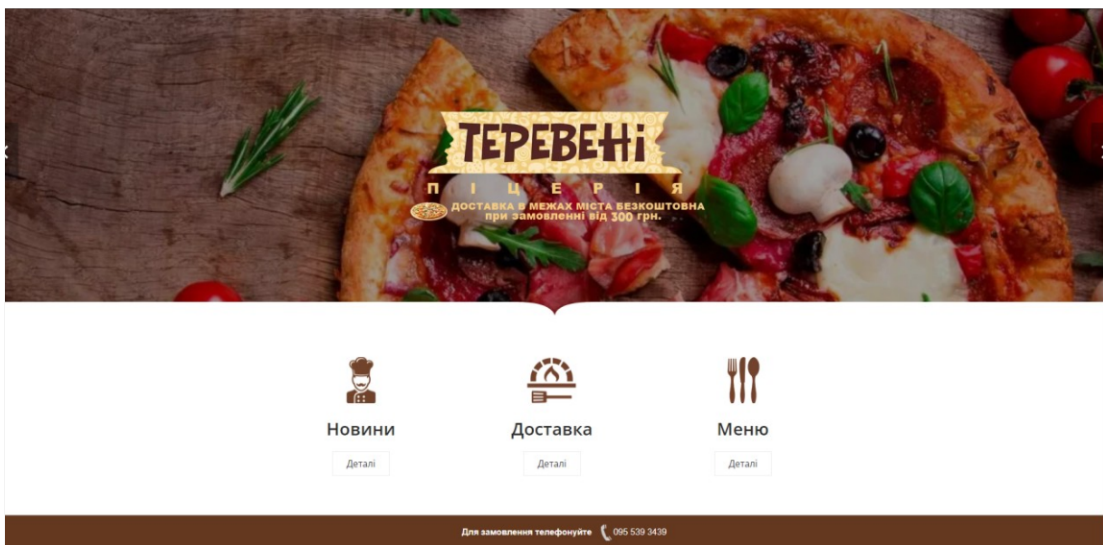


Рисунок 1.3 – Головна сторінка піцерії «Теревені»

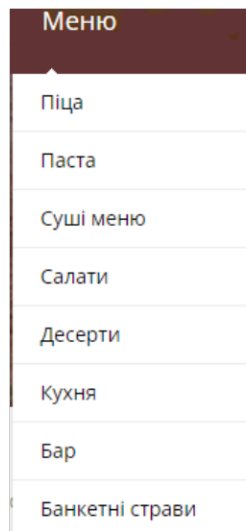


Рисунок 1.4 – Меню піцерії «Теревені»

Загалом, сайт піцерії «Теревені» доволі приємний в плані простоти навігації, та обраної палітри кольорів.

1.6.2 Веб-сайт піцерії «Pronto Pizza»

Pronto Pizza – один з найпопулярніших веб-сервісів для замовлення готової їжі у Луцьку. До переваг цього веб-сайта можна віднести великий вибір асортименту товарів, а також можливість самому обрати інгредієнти для піци у розділі Конструктор (Рис 1.5).

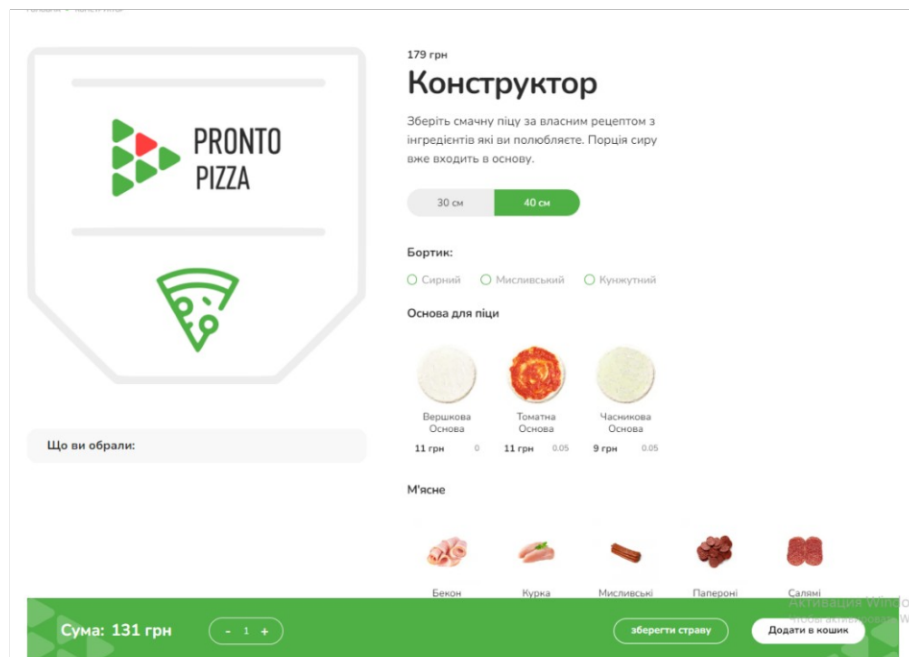


Рисунок 1.5 – Конструктор замовлення

Великою перевагою цього сайту є те, що клієнт може замовити їжу одразу, на сайті. Для того щоб це зробити, потрібно створити профіль (рис 1.6), після чого обрати потрібний вам товар, і одразу ж оплатити за допомогою банківської карти.

Зареєструватися

*Ваше ім'я

*Ваше прізвище

01.01.2000

Ваш телефон

*Введіть код

*Ваш емейл

Пароль

*Повторіть пароль

Я погоджуюсь з [Правилами користування](#)

Зареєструватись

На Ваш номер надіслано СМС повідомлення з підтвердженням телефону, введіть код вище.

Вже зареєстровані? [Увійти](#)

Рисунок 1.6 – Реєстраційна сторінка веб-сайту

Розглянувши всі переваги та недоліки проаналізованих сайтів, я вирішив, що у своїй кваліфікаційній роботі буду опиратися на веб-сайт піцерії «Теревені», адже мене зацікавила простота навігації цього, а також, простота його створення.

РОЗДІЛ 2

АНАЛІЗ ТА ВИБІР ЗАСОБІВ РОЗРОБКИ

2.1 Вибір середовища та інструментів розробки

Перш ніж починати розробку сайту, потрібно обрати текстовий редактор, або як кажуть ще, редактор коду. Розглядаючи редактори коду, я обрав Visual Studio Code, із-за таких його переваг перед іншими редакторами:

- має приємний вигляд;
- має автозаповнення синтаксису та гнучке налаштування;
- вбудовані Git команди;
- безліч розширень, серед яких є додаткові мови, інструменти та теми;
- VS Code є повністю безкоштовним текстовим редактором;

2.1.1 Огляд інтерфейсу VS Code

Щоб навчитися працювати з будь-якою програмою, в першу чергу, потрібно розібратися з її інтерфейсом (рис. 2.1). У VS Code досить простий інтерфейс.

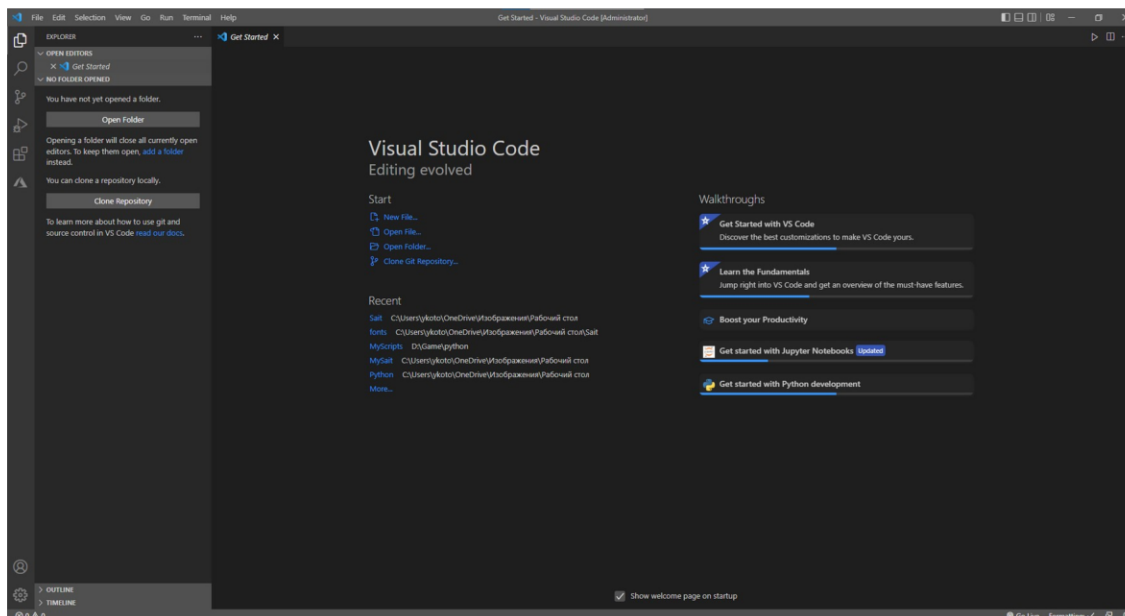


Рисунок 2.1 – Інтерфейс VS Code

Відкривши в перший раз VS Code, нам пропонують відкрити папку, створити новий файл, або ж вибрати файл, з яким ми будемо працювати. Також, на цій сторінці нам пропонують налаштувати наш редактор на мови програмування, які ми будемо використовувати, подивитись які є команди, налаштувати колір фону, а також обрати «гарячі клавіші», для швидкого виконання певних дій.

На лівій боковій панелі (рис 2.2), відображається шість основних вкладок, а саме:

- провідник – основна вкладка, за допомогою якої ми можемо відкрити потрібну нам папку, або файл;
- пошук – пошук заданого слова або словосполучення у файлах проекту;
- система контролю версіями – за допомогою цієї вкладки, ми можемо подивитися які зміни були внесені у проєкт;
- запуск та налагодження – у цій вкладці можна подивитись в якому стані знаходяться контрольні точки та змінні;
- розширення – у цій вкладці можна подивитися, які розширення встановлені, а також знайти потрібне для нас розширення і встановити;
- Майкрософт Ежур – хмарна платформа, призначена для розробників застосунків хмарних досліджень, спрощує процес створення онлайн-додатків.



Рисунок 2.2 – Бокова панель VS Code

2.1.2 HTML

HTML – це загальна мова розмітки документів для перегляду веб-сторінок у браузері. Для того, щоб вивести на екран інформацію із HTML файлу, браузери використовують, так звані, HTTP/HTTPS протоколи (ориг.

HyperText Transfer Protocol). Відмінність між HTTP та HTTPS протоколами у тому, що майже всі елементи, які є у HTTPS протоколах шифруються, а саме: URL-запити, параметри запиту, заголовки та куки. На сьогоднішній день, більшість веб-сайтів використовують саме HTTPS протоколи, для того, щоб гарантувати безпеку всім відвідувачам сайту, а також, браузери підвищують позиції у пошуковій видачі тих сайтів, які використовують HTTPS протоколи. Нажаль, є і недоліки у використанні цих протоколів, а саме: із-за шифрування особистих даних, збільшується навантаження на сервер, що в свою чергу, збільшує час завантаження сайту; HTTPS протоколи використовують SSL/TLS сертифікати, які платно затверджують спеціальні організації, є і безкоштовні сертифікати, але вони видаються на короткий термін часу, із-за чого їх потрібно часто замінювати.

Для того, щоб створити HTML файл, потрібно вказати його формат, а саме .html. Кожен HTML файл має свою загальну структуру, яка складається з трьох частин:

- декларації, в якій зазначається тип документу;
- шапки, яка відображає технічні відомості про файл, ця інформація не відображається в браузері;
- тіло документу, в якому відображається інформація про файл;

Щоб швидко створити структуру документу в VS Code, потрібно відкрити файл з форматом html, у якому поставити знак оклику (1 + "shift") і натиснути клавішу Enter (рис 2.3).

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10
11 </body>
12 </html>
```

Рисунок 2.3 — Загальна структура HTML файлу

Для того, щоб браузер відображав вашу інформацію на веб-сторінці, її потрібно записувати у тезі body, тег title визначає заголовок документа, який відображається у заголовку вікна браузера або на вкладці сторінки

За своїми властивостями елементи поділяються на типи:

– елементи структурної розмітки (рис 2.4) призначені для опису текстового контенту, до таких елементів належать теги: `<h>`; `<p>`; ``; тощо;



Рисунок 2.4 – Приклад елементів структурної розмітки

– елементи візуальної розмітки (рис. 2.5) використовують для візуальних ефектів, до таких тегів відносять: ``, `
`, ``, та інші;



Рисунок 2.5 – Приклад елементів візуальної розмітки

– елемент розмітки гіпертексту (рис 2.6), застосовується для посилання на інші веб-сайти, або веб-сторінки, прикладом такого тегу є `<a>`;



Рисунок 2.6 – Приклад гіпертекстової розмітки

Для того, щоб дізнатися докладну інформацію про різні теги, слід перейти за цим посиланням: css.in.ua/html/tags.

2.1.3 CSS

CSS (англ. Cascading Style Sheets) – це мова програмування, за допомогою якої можна описати загальний вигляд вебсторінки. Однією з її переваг є можливість розділити зміст сторінки, та візуально покращити її. Таке розділення може покращити сприйняття та доступ до контенту. Використовуючи CSS, один і той самий HTML документ можна по-різному відобразити. Таблицю стилів створюють в окремому файлі з розширенням .css. Щоб зв'язати CSS файл з HTML файлом (рис. 2.7), слід у тег <head> додати посилання на таблиці стилів, за допомогою тега <link>.

```

<head>
  <link rel="stylesheet" href="./Рабочий стол/index.css" />
</head>

```

Рисунок 2.7 –Зв'язування Кастадних таблиць з HTML файлом

Для опису стилю HTML тегів, застосовую певні правила, а саме: кожен тег повинен мати властивість і значення, які записуються у фігурних дужка, та розділяється двома крапками, у кінці значення завжди ставиться крапка з комою. Якщо у нас є два однакових теги, наприклад <div>, які ми часто будемо використовувати при написанні сайту, то потрібно вказувати для таких тегів різні класи, а починати описувати їх слід із крапки, після якої іде назва класу. Приклад опису стилю представлений у лістингу 2.1.

Лістинг 2.1 – Опис HTML тегів за допомогою CSS

```
h3 {
    color: red;
    font-family: cursive;
    font-size: 25px;
    text-align: center;
    border-style: none;
    border-width: 4px 1.3em;
}
.div1{
    background: #fff;
    border-radius: 10fr;
    width: 150px;
    float: right;
    padding: 12px;
}
.div2{
    width: 100px;
    border: solid 1px black;
    left: 10px;
    margin-left: 30px;
    background: rgb(133,53,53)
}
```

Кінець лістингу 2.1

2.1.4 LESS

Less – це динамічна мова програмування, яка інтерпретується в CSS. Цей препроцесор використовують для рівня абстракції коду та спрощення файлів каскадних таблиць, що дозволяє швидше та ефективніше писати код. Нажаль, в Visual Studio Code немає розширення для цієї мови, тому я вирішив використовувати кросплатформову програму для компіляції less коду – Koala. Основні переваги цієї програми:

- компіляція коду в реальному часі;
- підтримка кількох мов програмування, до яких, окрім less, відносяться: SASS, CSS, JS;
- налаштування компіляції;
- компресія;
- Повідомлення про помилки;
- кросплатформеність;

Інтерфейс програми Koala мінімалістичкий (рис. 2.8), і тому розібратися у ньому не виникає труднощів.

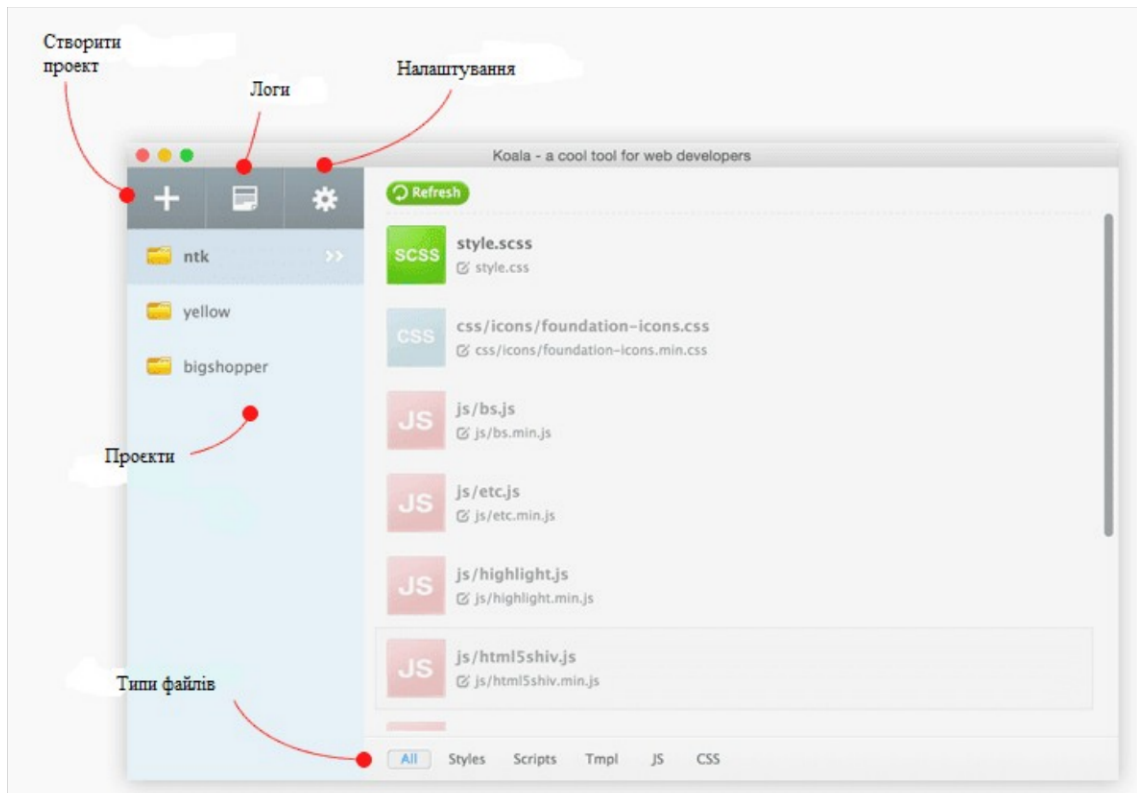


Рисунок 2.8 – Інтерфейс програми Koala

2.1.5 JavaScript

JavaScript – це динамічна, об’єктно-орієнтована мова програмування, яку у 1995 році розробив Брендан Айк для компанії NetScape. «Через рік, компанія вирішила розвивати JavaScript до промислового стандарту, в результаті чого, з’явилася стандартизована мова програмування ECMAScript,

перші три редакції були випущені упродовж 1997-1999 років. Наступні 10 років, нових редакцій не з'являлося, а у 2009 вийшла 5 редакція ECMAScript. Починаючи з 2015 року, комітет ECMAScript прийняв рішення, про щорічне оновлення версій. Останнє, одинадцяте, видання було опубліковане у 2020 році» [7].

На сьогоднішній день, JavaScript є найпопулярнішою мовою програмування в інтернеті, хоча так було не завжди. На початку свого існування, розробники її недооцінювали, та ставилися до неї скептично, але поява AJAX і її модифікацією до ES6+, яка внесла багато можливостей для швидкого та ефективного написання коду, її почали використовувати професійні спільноти.

«JavaScript є інтерпритованою мовою програмування, без строгої типізації, що змушує користувачів уважно перевіряти свій код в широкому переліку можливих конфігурацій» [8].

«JavaScript має декілька вбудованих об'єктів, які в свою чергу поділяються на групи: фундаментальні, помилки, числа та дати, текстові, індексовані, ключові, для роботи з структурованими даними, абстрактні та рефлексійні» [9].

Найчастіше ця мова використовується для створення веб-сторінок, крім цього вона використовується для:

- створення односторінкових та прогресивних вебзастосунків, за допомогою бібліотек React, AngularJS, Vue.js;
- програмування на боці сервера;
- стаціонарних та мобільних застосунків;
- сценаріїв у прикладних програмах, таких як Adobe Creative Suite, Apache JMeter.

Для того, щоб підключити JavaScript код до HTML файлу, застосовують спеціальний тег `<script>`. Підключати код можна у розділі `<head>` та `<body>`, хоча на мою думку, найкраще його підключати після закритого тегу `</body>`, для кращого сприйняття коду.

2.1.6 AJAX

AJAX (англ. Asynchronous JavaScript and XML) – це не самостійна мова програмування, це скоріше, концепція веб розробки, яку вперше публічно використав американський програміст, математик – Джессі Джеймсон Гаррет. Технологія Ајах використовується для обміну та оновлення даних із сервера, без оновлення всієї веб-сторінки. На сьогоднішій день, технологія Ајах поширена у всьому світі, адже його використовують для створення надзвичайно швидких та динамічних сторінок. Веб-сайти, які застосовують технологію Ајах у своїх додатках: Gmail, Google, Maps, FaceBook. Принцип роботи Ајах зображений на рисунку 2.9

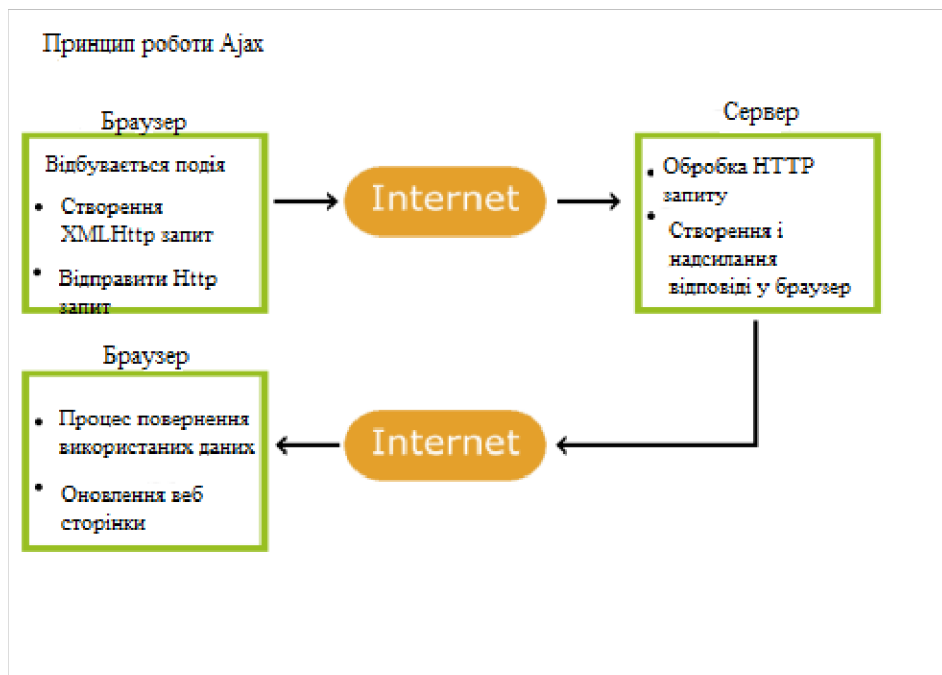


Рисунок 2.9 – Принцип роботи Ајах

2.1.7 JQuery

У 2006 році, американський інженер-програміст Джон Ресіг, створив і розробив бібліотеку JQuery для використання її разом із мовою програмування JavaScript. Це швидка, проста та лаконічна бібліотека, яка дозволяє писати JavaScript код набагато швидше і коротше. Мета JQuery – коротко написати код на JavaScript мові для створення динамічного сайту.

Преш ніж вивчати цю бібліотеку, слід добре розбиратися у таких мовах програмування як HTML, CSS та JavaScript. Маніпулювання HTML/DOM, CSS, методи подій HTML, ефекти та функції анімацій, AJAX, утиліти – всі ці інструменти використовують бібліотеку JQuery. Щоб підключити бібліотеку JQuery, її потрібно завантажити з офіційного сайту. JQuery поставляється з одним JavaScript і він записаний між HTML тегом <script>. Підключення бібліотеки наведено у рисунку 2.10

```
<head>  
<script src="jquery-1.12.0.min.js"></script>  
</head>
```

Рисунок 2.10 – Підключення бібліотеки JQuery

2.1.8 BootStrap

BootStrap – це відкритий, безкоштовний фреймворк для таких мов програмування, як HTML, CSS та JavaScript, який допомагає швидко створити дизайн вебсторінки. BootStrap є популярним фреймворком у світі, адже крім незалежних розробників, його, також, використовують великі компанії, як NASA та MSNBC.

Першою назвою цього фреймворку була Twitter Blueprint, і розроблялася вона для забезпечення однаковості внутрішніх інструментів для соціальної мережі Twitter.

До появи BS, у розробці інтерфейсу застосовувалися різні бібліотеки, які часто призводили до появи помилок, та ускладнювали оптимізацію. «Маленька група розробників, до складу якої входив і я, спроектувала і побудувала новий новий внутрішній інструмент, що мав потенціал для створення ширшого рішення, за кілька місяців з'явилася початкова версія BootStrap, яка є засобом для документування та поширення загальних шаблонів і засобів проектування всередині компанії, говорив один із розробників фреймворку Марк Отто» [10].

Переваги Bootstrap при розробці користувацької частини вебсайту:

- швидкість створення якісної верстки, навіть початковими веб-розробниками за рахунок використання готових класів і компонентів;
- кросбраузерність і кросплатформеність, що дозволяє коректно відображати роботу сайту у браузерах, які підтримують цю платформу;
- наявність великої кількості готових та продуманих компонентів;
- можливість налаштування під власний проєкт, за допомогою використання SCSS змінних;

2.2 Системи управління веб-контентом

Система управління вмістом – це програма, яка використовується для оновлення та підтримки веб-сайту. Це комп'ютерне програмне забезпечення, яке використовується для зміни, створення, модифікації та керуванням вмісту веб-сайту. Новинні статті, блоги та маркетингові брошури веб-сайту потрібно змінювати або часто оновлювати. CMS допомагає змінювати або оновлювати вміст веб-сайту. Відвідувачі сайту завжди отримують нову інформацію, і це змушує їх повторно переглядати веб-сайт. На сьогоднішній день, Drupal, WordPress та Joomla – є найпопулярнішими системами керування веб-контентом

2.2.1 WordPress

WordPress – це найпопулярніший веб-сайт системи керування вмістом, який допомагає створити та керувати веб-сайтом за допомогою різних інструментів. Без навичок програмування та дизайну, кожен може створити власний сайт за допомогою WordPress інструментів. Щоб запусити веб-сайт, на комп'ютер нічого не потрібно встановлювати. Для оновлення сайту достатньо лише підключення до Інтернету. Це програмне забезпечення з відкритим кодом, за допомогою якого, користувачі можуть обирати будь-який дизайн (відомий як «теми» у WordPress), а вибрані теми можна змінювати, оскільки вихідний код відповідає дизайну. WordPress.org є

офіційним веб-сайтом WordPress. Люди можуть використовувати безкоштовне доменне ім'я з WordPress.com або власне доменне ім'я, за яке потрібно буде заплатити. WordPress – це зручний спосіб підтримувати всю систему веб-сайту. Ось чому люди охоче використовують WordPress, і кількість користувачів також збільшується з кожним днем.

2.2.2 Joomla

Joomla – це сайт, який отримав нагороди серед інших систем керування контентом. Це безкоштовний відкритий код, за допомогою якого можна публікувати вміст для WWW (World Wide Web). Крім того, мова PHP і об'єктно-орієнтований стиль програмування (ООП) використовувалась для створення Joomla. Великою перевагою даної програми є те, що вона повністю безкоштовна, гнучка і проста у встановленні. Люди можуть легко використовувати інструменти Joomla у своїх потребах. Використання Joomla сьогодні зростає серед інших систем керування вмістом сайтів електронної комерції невеликих, а також, великих компаній.

2.2.3 Drupal

Drupal – це ще одна безкоштовна система керування вмістом з відкритим кодом, вона входить до трійки найвідоміших CMS. Персональні веб-сайти, блоги, соціальні сайти та сайти електронної комерції створені за допомогою Drupal. Велика і динамічна спільнота працює за системою Drupal. Вони постійно працюють над оновленням інструментів

Drupal. Люди легко отримують рішення для веб-сайту, пов'язане з вмістом, від спеціалістів і учасників Drupal

РОЗДІЛ 3

РОЗРОБКА ВЕБСАЙТУ ДЛЯ ТОРГІВЛІ НА ЗАМОВЛЕННЯ

3.1 Огляд головної сторінки веб-сайту

Головна сторінка веб-сайту – це найголовніша сторінка на веб-сайті, адже саме її бачать користувачі, які відвідують цей сайт вперше. Від оформлення і простоти орієнтування залежить, чи клієнт залишиться на вашому сайті, чи піде.

3.1.1 Заголовок веб-сайту

Велику роль у створенні вебсайту відіграє його заголовок (рис 3.1), адже саме там клієнт може дізнатися найголовнішу інформацію, а саме меню, доставку товару, спосіб оплати, контактні дані, а також, дійсні акцій. На відіну від тіла, заголовок, є незмінною частиною веб-сайту, він присутній на всіх сторінках. Лістинг заголовку, показано на рисунку 3.2, 3.3 та 3.4

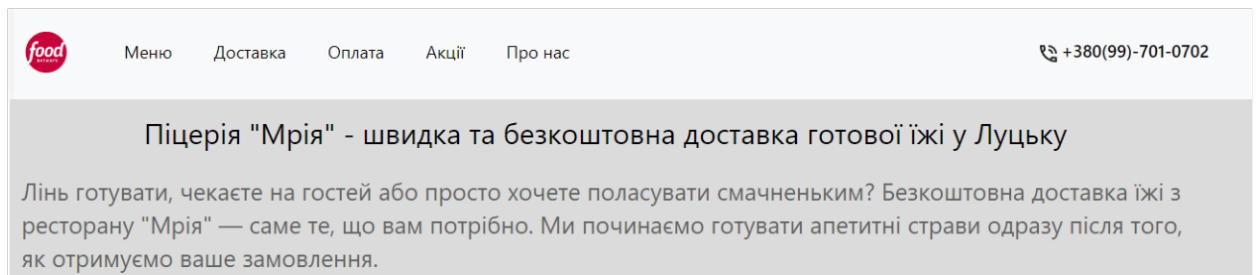


Рисунок 3.1 — Заголовок веб-сайту

```
<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <ul class="navbar-nav me-auto mb-2 mb-lg-0">
    <li class="nav-item dropdown">
      <a
        class="nav-link dropdown-toggle"
        href="#"
        role="button"
        data-bs-toggle="dropdown"
        aria-expanded="false"
      >
```

Рисунок 3.2 — Код заголовку HTML файлу

```

    Меню
  </a>
  <div class="dropdown-menu">
    <a class="dropdown-item" href="index.html">Супи</a>
    <a class="dropdown-item" href="pizza.html">Піци</a>
    <a class="dropdown-item" href="salad.html">Салати</a>
    <a class="dropdown-item" href="sushi.html">Суші</a>
    <a class="dropdown-item" href="pasta.html">Паста</a>
    <a class="dropdown-item" href="drinks.html">Напої</a>
  </div>
</li>
<li class="nav-item active">
  <a
    class="nav-link active"
    aria-current="page"
    href="./page.html"
  >Доставка</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="./order.html">Оплата</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="./Actions.html">Акції</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="./aboutus.html">Про нас</a>
</li>
</ul>
<div class="contact-information d-flex">
  <div class="phone">
    <a href="tel:+380997010702"></a>
    <a href="">
      
      <span>+380(99)-701-0702</span>
    </a>
  </div>

```

Рисунок 3.3 – Код HTML файлу

```

nav {
  width: 100%;
  height: 100px;
}
.dropdown-menu {
  background: #ffffff;
  box-shadow: 0px 0px 15px rgba(0, 0, 0, 0.12);
  border-radius: 12px;
  border: none;
}
.navbar-expand-lg .navbar-nav .nav-link {
  font-style: normal;
  font-weight: 500px;
  font-size: 18px;
  line-height: 22px;
  color: #000000;
  padding: 0 22px;
  &:hover {
    color: #00ac4e;
  }
}

```

Рисунок 3.4 – Код CSS файлу

3.1.2 Тіло веб-сторінки

У тілі веб-сторінки, переважно, знаходиться основна інформація, яку потрібно донести клієнтові. Так як головна мета мого сайту – продаж товарів, то основна інформація, яка знаходиться на сторінках цього веб-сайту є асортимент торавір, який зображений на рисунку 3.5. Частина коду зображена на рисунках 3.6 та 3.7.

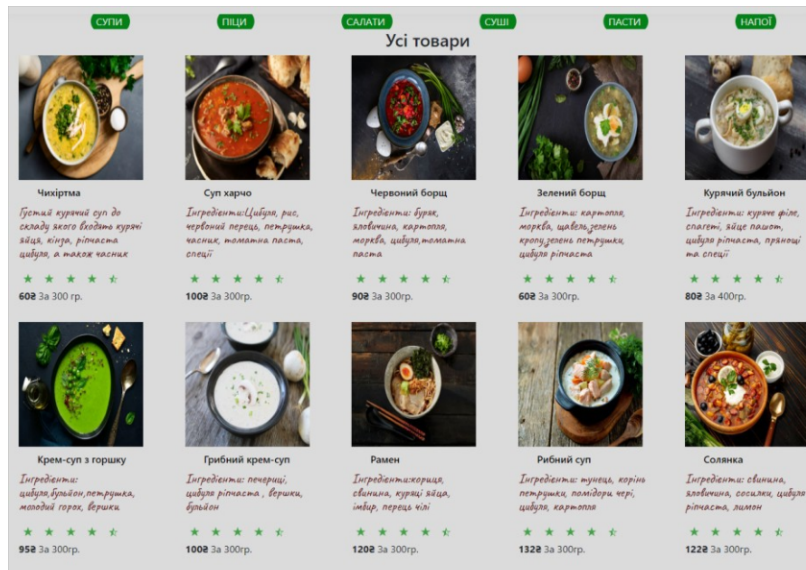


Рисунок 3.5 – Розділ «Супи»

```

</div>
<a href="/Chihirtma.html" class="product-title">Чихіртма</a>
<p class="product-description">
Густий курячий суп до складу якого входять курчі яйця, кінза, річчаста цибуля, а також часник
</p>
<div class="price"><b>60€</b> <span>За 300 гр.</span></div>
</div>
<div class="product">
<div class="img">

</div>
<a href="/Harcho.html" class="product-title">Суп харчо</a>
<p class="product-description">
Інгредієнти:Цибуля, рис, червоний перець, петрушка, часник, томатна паста, <br /> спеції
</p>
<div class="price"><b>100€</b> <span>За 300гр.</span></div>
</div>
<div class="product">
<div class="img">

</div>
<a href="/redborshch.html" class="product-title">Червоний борщ</a>
<p class="product-description">
Інгредієнти: буряк, яловичина, картопля, морква, цибуля, томатна паста
</p>
<div class="price"><b>90€</b> <span>За 300гр.</span></div>
</div>

```

Рисунок 3.6 — HTML код тіла веб-сторінки

```

.contact-information {
  align-items: center;
  .phone {
    margin-right: 35px;
    a {
      font-weight: 500;
      font-size: 18px;
      line-height: 21px;
      color: #000000;
      text-decoration: none;
      .img {
        margin-right: 5px;
      }
    }
  }
}
.menu-user {
  .cart-content {
    display: none;
  }
  a {
    margin-left: 35px;
    float: right;
    cursor: pointer;
    transition: color 500ms ease;
    &:hover {
      color: red;
    }
  }
}

```

Рисунок 3.7 – CSS код тіла веб-сторінки

3.1.3 Footer веб-сайту

Фу́тер сайту або, як ще кажуть, підвал – це блок з другорядною інформацією, він знаходиться в нижній частині веб-сторінки. В ньому, переважно, знаходиться посилання на інші матеріали ресурсу, посилання на соціальні мережі, контактна інформація, графік роботи.

Footer - це незмінний елемент сайту, так само як і хедер. Він допомагає перелінкувати між собою сторінки порталу, природним чином підвищивши залученість користувачів. У ньому, зазвичай, знаходиться інформація, яку можливо легко знайти на будь-якій сторінці цього сайту, тому в ньому може розміщуватись інформація, яка знаходиться в хедері. Інформація, яка знаходиться у футері веб-сторінки залежить від типу сайту, тематики,

дизайну, змісту інших блоків, тощо. У нижній частині сайту вказують копірайт, умови при яких дозволено використовувати та поширювати інформацію, яка знаходиться на цьому сайті, інформацію та контактні дані розробників порталу, конфіденційність, способи оплати та інша необхідна інформація для полегшення користування веб-сервісом клієнтом. Крім цього, у цій частині веб-сайту може знаходитись логотип компанії, можливі способи оплати та вказані додаткові сторінки. Футер веб-сайту зображений на рисунку 3.8. Лістинг програмної частини зображений на рисунку 3.9 та 3.10

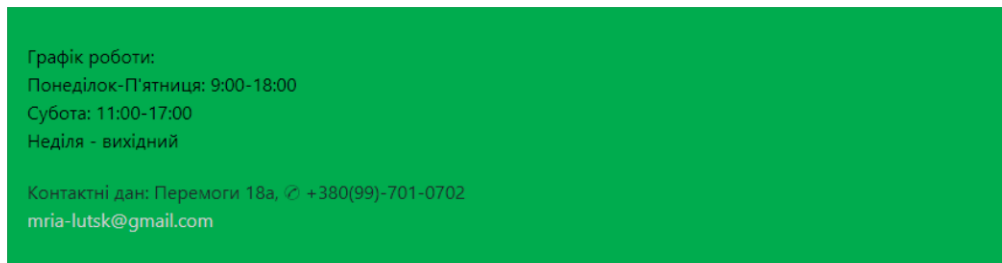


Рисунок 3.8 – Колонтитул веб-сайту

```
<footer>
  <div class="container">
    <div class="footer-wrapper">
      <div class="footer-left">
        <div class="phones">
          <div class="work-time">
            Графік роботи: <br />
            Понеділок-П'ятниця: 9:00-18:00 <br />
            Субота: 11:00-17:00 <br />
            Неділя - вихідний
          </div>
        </div>
      </div>
      <div class="footer-center">
        <div class="social">
          Контактні дан: Перемоги 18а, ☎ +380(99)-701-0702
        </div>
        <div class="work-email">
          <a
            class="email"
            >mria-lutsk@gmail.com</a>
        </div>
      </div>
    </div>
  </div>
</div>
```

Рисунок 3.9 – HTML код колонтитула

```

.to-pay {
  span {
    font-size: 500;
    font-weight: 14.45px;
    line-height: 22px;
    color: #000000;
  }
  b {
    margin-left: 90px;
    font-weight: 500;
    font-size: 18px;
    line-height: 23px;
    color: #00ac4e;
    font-style: normal;
  }
}

.pay-block {
  .pay-popup {
    margin-top: 20px;
    width: 194px;
    height: 40px;
    left: 935px;
    top: 515px;
    background: #00ac4e;
    border-radius: 4.5px;
    display: flex;
    justify-content: center;
    align-items: center;
  }
}

```

Рисунок 3.10 – CSS код колунтитула

3.2 Навігація на веб-сайті

На веб-сайті головне меню може мати форму як горизонтальних наскрізних меню, так і вертикальних бічних меню. Обидва варіанти найчастіше використовуються у великих магазинах.

Послідовне головне меню. Меню необхідні для швидкого переходу від розділу до розділу. Інтернет-магазини із середньою кількістю товарів зазвичай не можуть включити всі типи товарів у таке меню.

Наскрізнi меню можуть бути як горизонтальними, так і вертикальними бічними панелями. Горизонтальні головні меню частіше зустрічаються у великих магазинах, де зазвичай розміщуються основні категорії вищого рівня.

Меню довідки. Роль довідкового меню полягає у відображенні підкатегорій, якщо користувач вже визначився з типом товару. Наприклад, вкажіть, які продукти доступні на вашому веб-сайті. Інтернет-магазини часто замінюють додаткові меню фільтрами. Фільтри зручні та знайомі користувачам і допомагають уникнути складних і заплутаних внутрішніх елементів. На сайті інтернет-магазину реалізовано.

3.3 Технічні вимоги до середовищ використання

«Мінімальними вимогами для процесорів, які запускаються в браузер на операційній системі Windows є Internet Explorer 7, Firefox 3.5, Safari 3.2.1, або Chrome 2. Він потребує 233 Mhz процесор; мінімум 64 MB RAM (для Windows XP), рекомендовано 512 MB RAM; мінімум 150 MB (Windows XP), 120 MB (Windows Vista) вільного місця на диску» [11].

«Для реалізації статичних сторінок, а також, шаблонів повинні використовуватися мови програмування HTML 5.0, CSS та Less. Вихідний код повинен розроблятися відповідно до стандартів W3C (HTML 5.0). Для реалізації інтерактивних елементів клієнтської частини повинні використовуватися мови JavaScript» [12].

3.4 Принцип роботи веб-сайту

Принцип роботи веб-сайту полягає в наступному: користувач заходить на наш сайт, після чого бачить головну сторінку, на якій він має можливість переглянути ознайомитись з меню товарів, оглянути дійсні акції та дізнатись

інформацію про графік роботи. Після того, як клієнт обрав потрібний йому товар, він може його замовити, зателефонувавши на гарячу лінію.

ВИСНОВКИ

У ході виконання даної кваліфікаційної роботи було проведено детальне дослідження веб-сайтів, які спеціалізуються на продажі готової продукції на замовлення, із всіх сайтів, які я оглянув, найбільше мені сподобався веб-сервіс піцерії «Теревені». Основними перевагами цього сайту були: простота створення, міжсторінкова навігація та простий, але стильний дизайн.

В ході дослідження були проаналізовані основні етапи веб-розробки сайтів, а саме: прототипування, розробка дизайну, програмування та тестування. Крім цього, було детально проаналізовано і обґрунтовано вибір інструментів, за допомогою яких було створено сайт: HTML, CSS, JS, а також бібліотеку JQuery.

У процесі реалізації було створено веб-сайт, завдання якого полягало у тому, що будь-який користувач міг обрати, а також, замовити потрібний йому товар із запропонованого асортименту.

Після реалізації було проведено тестування розробленого об'єкту для перевірки його функціональності та надійності роботи, розроблено інструкцію з використання, яка дозволить користувачам швидко ознайомитися з навігацією на веб-сайті.

Завдяки використанню сучасних мов програмування та провідних бібліотек, гарантується підтримка та подальший розвиток усіх компонентів програмної системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мацегора, М. В. Засоби моніторингу енергоефективних та відновлювальних джерел енергії: випускна кваліфікаційна робота: 123 "Комп'ютерна інженерія" / М. В. Мацегора ; керівник роботи О. В. Тарасенко-Клятченко ; НУ "Київська політехніка", кафедра інформаційних та комп'ютерних систем. Київ, 2019. 82 с.

2. Морозюк, М. В. Методи та засоби розробки web-сайту інтернет-магазину: випускна кваліфікаційна робота: 186 "Видавництво та поліграфія" / М. В. Морозюк ; керівник роботи Н. М. Шабицька ; НУ "Чернігівська політехніка", кафедра комп'ютерних мультимедійних технологій. Чернігі, 2020. 92 с.

3. Less Logical Function. LessCSS. URL: <https://lesscss.org/functions/> (дата звернення: 13.01.2023).

4. Тимошенко К. О. Сучасні інструменти JS-розробника. Кропивницький: ЦНТУ, 2020. 12 с.

5. Середня вартість різних веб-рішень. SCnsoft. URL: <https://www.scnsoft.com/services/web-development> (дата звернення: 17.01.2023)

6. HTML. Вікіпедія. URL: <https://uk.wikipedia.org/wiki/HTML> (дата звернення: 23.01.2023)

7. CSS Helper Classes Reference. W3Schools. URL: https://www.w3schools.com/bootstrap/bootstrap_ref_css_helpers.asp (дата звернення: 02.02.2023)

8. Кортельова, Я. О. Геометричне моделювання в інформаційних системах: кваліфікаційна робота освітнього рівня «Бакалавр». Київ: КПІ, 2020. 66 с.

9. Drouglas Crockford's JavaScript. Crockford's. URL: <http://crockford.com/javascript/> (дата звернення: 22.02.2023).

10. Bootstrap Documentation. Bootstrap. URL: <https://getbootstrap.com/docs/5.3/getting-started/introduction/> (дата звернення: 26.02.2023).

11. Bootstrap. Вікіпедія. URL: <https://uk.wikipedia.org/wiki/Bootstrap> (дата звернення: 02.03.2023).

12. JQuery Tutorial. JQuery. URL: <https://jquery.com/forum> (дата звернення: 06.03.2023).

13. Most popular JS stack. Medium. URL: <https://medium.datadriveninvestor.com/most-popular-technology-stack-to-choose-from-full-stack-vs-mean-stack-vs-mern-stack-in-2019-d12c0a17439a> (дата звернення: 09.03.2023).

ДОДАТКИ

Додаток А

Лістинг коду файлу index.html

```

<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Піцерія 'Мрія'</title>
    <link rel="preconnect" href="https://fonts.googleapis.com" />
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
    <link
      href="https://fonts.googleapis.com/css2?
family=Caveat&family=Montserrat:ital,wght@0,700;1,400&display=swap"
      rel="stylesheet"
    />
    <link rel="stylesheet" href="style.css" />
    <link rel="stylesheet" href="fonts/front.css" />
    <link
      href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/
bootstrap.min.css"
      rel="stylesheet"

integrity="sha384-GLhlTQ8iRABdZLl6O3oVMWSktQOp6b7In1Zl3/Jr59b6EGGoI1aFkw7cmDA
6j6gD"
      crossorigin="anonymous"
    />
    <script
src="http://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"></
script>
  </head>

  <body>
    <section class="wrapper">
      <header>
        <div class="contauener">
          <nav class="navbar navbar-expand-lg bg-body-tertiary">
            <div class="container-fluid">
              <a class="navbar-brand" href="./index.html"
                ></a>
              <button
                class="navbar-toggler"
                type="button"
                data-bs-toggle="collapse"
                data-bs-target="#navbarSupportedContent"
                aria-controls="navbarSupportedContent"
                aria-expanded="false"
                aria-label="Переключатель навигации"
              >
                <span class="navbar-toggler-icon"></span>
              </button>
              <div class="collapse navbar-collapse"
id="navbarSupportedContent">
                <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                  <li class="nav-item dropdown">
                    <a
                      class="nav-link dropdown-toggle"
                      href="#"
                      role="button"
                      data-bs-toggle="dropdown"
                      aria-expanded="false"

```

>

```

        Меню
    </a>
    <div class="dropdown-menu">
        <a class="dropdown-item" href="index.html">Супи</a>
        <a class="dropdown-item" href="pizza.html">Піци</a>
        <a class="dropdown-item" href="salad.html">Салати</a>
        <a class="dropdown-item" href="sushi.html">Суші</a>
        <a class="dropdown-item" href="pasta.html">Паста</a>
        <a class="dropdown-item" href="drinks.html">Напої</a>
    </div>
</li>
<li class="nav-item active">
    <a
        class="nav-link active"
        aria-current="page"
        href="/page.html"
    >Доставка</a>
    >
</li>
<li class="nav-item">
    <a class="nav-link" href="/order.html">Оплата</a>
</li>
<li class="nav-item">
    <a class="nav-link" href="/Actions.html">Акції</a>
</li>
<li class="nav-item">
    <a class="nav-link" href="/aboutus.html">Про нас</a>
</li>
</ul>
<div class="contact-information d-flex">
    <div class="phone">
        <a href="tel:+380997010702"></a>
        <a href="">
            
            <span>+380 (99) -701-0702</span>
        </a>
    </div>
</div>
</div>
</nav>
</div>
</header>
<div class="home-banner">
    <div class="contauner">
        <div class="row">
            <div class="block">
                <h2>
                    Піцерія "Мрія" - швидка та безкоштовна доставка готової їжі у
                    Луцьку
                </h2>
                <p>
                    Лінь готувати, чекаєте на гостей або просто хочете поласувати
                    смачненьким? Безкоштовна доставка їжі з ресторану "Мрія" –
                    саме
                    те, що вам потрібно. Ми починаємо готувати апетитні страви
                    одразу після того, як отримуємо ваше замовлення.
                </p>
            </div>
        </div>
    </div>
</div>
</div>
<section>
    <div class="contauner">

```

```

<div class="swiper-container">
  <a href="index.html" class="swiper-slide active">
    <span>СУПИ</span>
  </a>
  <a href="pizza.html" class="swiper-slide">
    <span>ПИЦЦИ</span>
  </a>
  <a href="salad.html" class="swiper-slide">
    <span>САЛАТИ</span>
  </a>
  <a href="sushi.html" class="swiper-slide">
    <span>СУШИ</span>
  </a>
  <a href="pasta.html" class="swiper-slide">
    <span>ПАСТИ</span>
  </a>
  <a href="drinks.html" class="swiper-slide">
    <span>НАПОЇ</span>
  </a>
</div>
</div>
</section>
</section>
<section class="home-products">
  <div class="container">
    <div class="row">
      <h3 class="all_products">Усі товари</h3>
      <div class="products-wrapper">
        <div class="product">
          <div class="price"><b>60€</b> <span>За 300 гр.</span></div>
        </div>
        <div class="product">
          <div class="img">
            
          </div>
          <a href="harcho.html" class="product-title">Суп харчо</a>
          <p class="product-description">
            Інґредієнти: Цибуля, рис, червоний перець, петрушка, часник,
            томатна паста, <br />
            спеції
          </p>
          <div class="price"><b>100€</b> <span>За 300гр.</span></div>
        </div>
        <div class="product">
          <div class="img">
            
          </div>
          <a href="redborsch.html" class="product-title">Червоний
борщ</a>
          <p class="product-description">
            Інґредієнти: буряк, яловичина, картопля, морква,
цибуля, томатна
            паста
          </p>

```

```
<div class="price"><b>90€</b> <span>3a 300rp.</span></div>
```

```

</div>
<div class="product">
  <div class="img">
    
  </div>
  <a href="zelenyborsch.html" class="product-title">Зелений
борщ</a>
  <p class="product-description">
    Інґредієнти: картопля, морква, щавель, зелень кропу, зелень
    петрушки, <br />
    цибуля ріпчаста
  </p>

  <div class="price"><b>60€</b> <span>За 300гр.</span></div>
</div>
<div class="product">
  <div class="img">
    
  </div>
  <a href="kurbulion.html" class="product-title">Курячий
бульйон</a>
  <p class="product-description">
    Інґредієнти: куряче філе, спаґеті, яйце пашот, цибуля
    ріпчаста,
    прянощі та спеції
  </p>

  <div class="price"><b>80€</b> <span>За 400гр.</span></div>
</div>
<div class="products-wrapper">
  <div class="product">
    <div class="img">
      
    </div>
    <a href="krem-sup-goroh.html" class="product-title"
    >Крем-суп з горшку</a>
    >
    <p class="product-description">
      Інґредієнти: цибуля, бульйон, петрушка, молодий горох, вершки
    </p>

    <div class="price"><b>95€</b> <span>За 300гр.</span></div>
  </div>
  <div class="product">
    <div class="img">
      
</div>
<a href="krem-sup-grub.html" class="product-title"
>Грибний крем-суп</a
>
<p class="product-description">
    Інґредієнти: печериці, цибуля ріпчаста , вершки, бульйон
</p>

    <div class="price"><b>100€</b> <span>За 300гр.</span></div>
</div>
<div class="product">
    <div class="img">
        
    </div>
    <a href="ramen.html" class="product-title">Рамен</a>
    <p class="product-description">
        Інґредієнти:кориця, свинина, куряці яйца, імбир, перець чілі
    </p>
    <div class="price"><b>120€</b> <span>За 300гр.</span></div>
</div>
<div class="product">
    <div class="img">
        
    </div>
    <a href="rub-sup.html" class="product-title">Рибний суп</a>
    <p class="product-description">
        Інґредієнти: тунець, корінь петрушки, помідори чері, цибуля,
        картопля
    </p>
    <div class="price"><b>132€</b> <span>За 300гр.</span></div>
</div>
<div class="product">
    <div class="img">
        
    </div>
    <a href="solianka.html" class="product-title">Солянка</a>
    <p class="product-description">
        Інґредієнти: свинина, яловичина, сосилки, цибуля ріпчаста,
ЛИМОН
    </p>
</div>
<div class="pagination-block">
    <nav aria-label="Page navigation example">
        <ul class="pagination">
            <li class="page-item">
                <a class="page-link" href="#" aria-label="Previous">
                    <span aria-hidden="true">&laquo;</span>
                </a>
            </li>
            <li class="page-item active">
                <a class="page-link" href="#">1</a>
            </li>
            <li class="page-item">
                <a class="page-link" href="pizza.html">2</a>
            </li>
        </ul>
    </nav>
</div>

```

```
<li class="page-item">
  <a class="page-link" href="salad.html">3</a>
</li>
<li class="page-item"><a class="page-link"
href="#">...</a></li>
<li class="page-item">
  <a class="page-link" href="drinks.html">6</a>
</li>
<li class="page-item">
  <a class="page-link" href="#" aria-label="Next">
    <span aria-hidden="true">&raquo;</span>
  </a>
</li>
</ul>
</nav>
</div>
</div>
</div>
</body>
</html>
```

Додаток Б**Лістинг коду файлу style.less**

```
body {
  font-family: "TT Norms";
  font-size: 16px;
  background-color: #ca3838;
}
header {
  margin-left: 300px;
  margin-right: 300px;
  background: 100px;
  box-shadow: 0px 0px 15px rgba(0, 0, 0, 0.12);
  background-color: #6c8d83;
  .navbar-brand {
    margin-right: 38px;
  }
  nav {
    width: 100%;
    height: 100px;
  }
  .dropdown-menu {
    background: #ffffff;
    box-shadow: 0px 0px 15px rgba(0, 0, 0, 0.12);
    border-radius: 12px;
    border: none;
  }
  .navbar-expand-lg .navbar-nav .nav-link {
    font-style: normal;
    font-weight: 500px;
    font-size: 18px;
    line-height: 22px;
    color: #000000;
    padding: 0 22px;
    &:hover {
      color: #00ac4e;
    }
  }
  .dropdown-toggle::after {
    display: none;
  }
  .contact-information {
    align-items: center;
  }
```

```
.phone {
  margin-right: 35px;
  a {
    font-weight: 500;
    font-size: 18px;
    line-height: 21px;
    color: #000000;
    text-decoration: none;
    .img {
      margin-right: 5px;
    }
  }
}

.menu-user {
  .cart-content {
    display: none;
  }
  a {
    margin-left: 35px;
    float: right;
    cursor: pointer;
    transition: color 500ms ease;
    &:hover {
      color: red;
    }
  }
  button {
    margin-left: 30px;
  }
  button {
    background: none;
    border: none;
    padding: 0;
  }
}

.home-banner {
  .block {
    color: #ffa6cb;
    margin-left: 16%;
    background: #dbdbdb;
    background-repeat: no-repeat;
  }
}
```

```
background-position: top right;
box-shadow: 0px 0px 18px rgba(0, 0, 0, 0.03);
border-radius: 1px;
height: 200px;
width: 1310px;
overflow: hidden;
background-size: auto 100%;
h2 {
  font-style: normal;
  font-weight: normal;
  font-size: 30px;
  line-height: 70px;
  color: #000000;
  margin-left: 10%;
}
p {
  margin-right: 40px;
  font-size: 25px;
  line-height: 35px;
  color: #5e6462;
}
a {
  &:hover {
    color: #727272;
  }
  margin-left: 190px;
  font-weight: 500;
  font-size: 30px;
  line-height: 30px;
  color: #000000;
  text-decoration: none;
  background: #0fc530;
  border-radius: 10px;
  padding: 5px 8px;
}
img {
  margin-left: 480px;
}
}
.swiper-slide {
  flex-shrink: 0;
  width: 100%;
  height: 100%;
```

```

    position: relative;
    transition-property: transform;
  }
}
.contauner {
  .swiper-container {
    border-radius: 1px;
    box-shadow: 0px 0px 18px rgba(0, 0, 0, 0.03);
    background-position: top right;
    background-repeat: no-repeat;
    overflow: hidden;
    width: 68.8%;
    background: #dbdbdb;
    margin-left: 15.57%;
    .swiper-slide {
      margin-left: 140px;
      text-decoration: none;
      color: #ebebeb;
      border-radius: 20px;
      padding: 8px 8px;
      background: #048219;
    }
  }
}
}
.home-products {
  margin-left: 310px;
  .row {
    .pagination-block {
      display: flex;
      justify-content: center;
      ul {
        li {
          a {
            width: 41px;
            height: 41px;
            font-size: normal;
            font-weight: 500px;
            font-style: 20px;
            line-height: 25px;
            color: #a12121;
            background: #dbdbdb;
            border-radius: 5px;
            text-align: center;
          }
        }
      }
    }
  }
}

```

```
    }
    &.active {
      a {
        background: #00ac4e;
        color: #dbdbdb;
      }
    }
  }
}
flex-direction: column;
background: #dbdbdb;
width: 82.26%;
.all_products {
  margin-left: 600px;
  background: #dbdbdb;
  width: 160px;
}
}
}
```