

**ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ,
ПРАВА ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

Пояснювальна записка

до кваліфікаційної роботи на здобуття ступеня вищої освіти бакалавр
на тему: «Проектування комерційного промо-сайту з використанням
функцій та бібліотек Java Script»

Виконав: здобувач вищої освіти
за освітньо-професійною
програмою Інформаційні
управляючі системи спеціальності
126 Інформаційні системи та
технології
освітнього ступеня бакалавр
групи 126ІСТ_бд_2022[1](стн)
Куляєв Р.Є.
Керівник: Копішинська О.П.
Рецензент: Муравльов В.В.

Полтава – 2024 року

ВСТУП

Всі ефективні засоби представлення та поширення інформації на сьогодні є високо затребуваними у всіх сферах життя, особливо у бізнесі. Аудиторія мережі інтернет збільшується з кожним роком. Не зважаючи на величезну кількість вебсайтів, які розміщені і функціонують в мережі інтернет, потреба в таких застосунках постійно зростає. Адже володіння інформацією й уміння її обробляти, а також швидко й направлено доносити її до клієнтів – ключ до успіху і зростання. Користувачі мережі є важливою ціллю для рекламодавців як потенційні клієнти, а вдалий комерційний вебсайт – це ефективний інструмент, здатний привернути увагу різної аудиторії.

Актуальність теми кваліфікаційної роботи розглядається у контексті важливості інформаційного супроводу діяльності будь-якої бізнес-компанії або організації, який забезпечується вебдодатками. Сучасний динамічний промо-сайт для малого та середнього бізнесу розглядається як важливий ресурс розгортання та здійснення підприємницької, комерційної, управлінської діяльності, засіб ефективних комунікацій та зручний додаток для кожного користувача.

Метою кваліфікаційної роботи є практична реалізація знань і вмінь застосування сучасних вебтехнологій для проектування й розробки промо-сайту для бізнесу.

Завдання кваліфікаційної роботи: здійснити ґрунтовний аналіз і порівняння існуючих засобів проектування і розробки вебзастосунків; з'ясувати відмінні характеристики промо-сайтів від інших типів; розробити макет дизайну і структури комерційного промо-сайту для компанії на прикладі обраної сфери діяльності; забезпечити зручний інтерфейс користувача та функціональність, оптимальність коду за допомогою відповідних технічних засобів, зокрема бібліотек мови програмування Java Script; здійснити попередній технічний аудит, економічне оцінювання вартості робіт та ефективності впровадження.

Об'єктом дослідження кваліфікаційної роботи є процеси розроблення дизайну та верстки промо-сайту підприємства малого бізнесу.

Предметом дослідження є підбір та композиція відповідних вебтехнологій, середовищ проєктування для розроблення сучасного вебсайту.

Методи дослідження: інформаційно-пошуковий, аналітико-синтетичний, емпіричний, порівняльний, абстрактний, графічний, доставки готових кодів та інші.

Інформаційну базу кваліфікаційної роботи складають: наукові статті та публікації в інтернет-виданнях, тематична література, навчальні відеоролики, статистичні дані аналітичних компаній, які розміщені на вебсайтах у вільному доступі, нормативні документи в галузі ІТ, стандарти тощо.

Практична значущість роботи: розроблення промо-сайту для ведення бізнесу в області ІТ є внеском у власне портфоліо розробок, а також може бути запропоновано для використання і ведення багатьох видів підприємництва невеликими компаніями, демонстрації й використання окремих інструментів розробника в навчальному процесі при вивченні вебтехнологій.

Структура і обсяг кваліфікаційної роботи: пояснювальна записка має 49 сторінок і складається з переліку скорочень, вступу, трьох розділів; висновків, списку використаних джерел, додатків. Робота містить 31 рисунок, 4 таблиці, посилання на 40 інформаційних джерел.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ОСНОВИ ЗАСТОСУВАННЯ БАЗОВИХ ТЕХНОЛОГІЙ ДЛЯ ПРОЄКТУВАННЯ ВЕБСАЙТІВ

1.1 Технічні та комунікаційні передумови перетворення вебсайтів у домінуючу форму представлення інформації в мережі інтернет

Історично перший вебсайт як адресоване місце для представлення і доступу інформації з'явився 20 грудня 1990 р., де було розміщено опис інноваційної технології World Wide Web (скорочено W3), заснованої на протоколі передачі даних HTTP, системі адресації URL та мові гіпертекстової розмітки HTML (HyperText Markup Language) [1].

Відомим є факт, що мова гіпертекстової розмітки HTML була розроблена групою вчених на чолі з британцем Тімом Бернерсом-Лі у 1986-1991 рр. у стінах Центру ядерних досліджень (CERN) у Женеві (Швейцарія), і від початку створювалася як мова для обміну технічною та науковою документацією, яку б могли використовувати люди, які не є фахівцями в області верстання [2].

Вебсайт у контексті первинного задуму – це набір файлів і документів, які поєднані спеціальними зв'язками – гіперпосиланнями, знаходяться на якомусь сервері і об'єднані під одне доменне ім'я. Вебсайт для бізнесу виконує важливі функції: реклама послуг, ідей, продукції; продаж товарів, послуг, ідей; безкоштовне надання інформації; підтримка клієнтів та багато інших [3].

Винайдення технологій розробки сайтів стало, по суті, революційною подією світового масштабу. Стандартами та специфікаціями ключових технологій розроблення вебсайтів займається W3C-консорціум, який від початку і до сьогодні очолює легендарний британець Тім Бернес-Лі. Протягом вже майже сорока років мова гіпертексту HTML розвивається і вдосконалюється завдяки відкритості платформи W3C [4]. Всі специфікації, пояснення, приклади розробники можуть знайти на офіційному вебсайті консорціуму. Стандарти W3C визначають відкриту вебплатформу для

розробки додатків, яка має безпрецедентний потенціал, щоб дозволити розробникам створювати насичений інтерактивний досвід на основі величезних сховищ даних, доступних на будь-якому пристрої [5]. Хоча межі платформи продовжують розвиватися, лідери галузі говорять майже в унісон про те, що HTML5 стане наріжним каменем для цієї платформи. Але повна потужність платформи спирається на багато інших технологій, які створюють W3C та її партнери, включаючи CSS, SVG, WOFF, стек семантичного вебу, XML та різноманітні API. W3C розробляє ці технічні специфікації та вказівки за допомогою процесу, розробленого для досягнення максимального консенсусу щодо змісту технічного звіту, забезпечення високої технічної та редакційної якості та отримання схвалення W3C та ширшої спільноти.

Вебсайт складається із пов'язаних між собою вебсторінок. Вебсторінка представляє собою текстовий файл із розширенням *.html, який містить текст, який буде зображено на сайті, а також HTML теги, які й визначають, як буде розміщуватись інформація на сторінці. HTML документи зручні тим, що для їх перегляду підходить майже будь-який браузер [6].

Доменне ім'я – це унікальна адреса сайту, яка використовується для його специфікації та пошуку в мережі. Воно складається з літер та цифр, а також може містити символ «-». Мінімальна довжина – 1 символ, максимальна – 63 символи [7]. Наприклад, «www.pdau.edu.ua» є доменним ім'ям Полтавського державного аграрного університету.

Домен першого рівня – це права частина домену після крапки. Цю частину може ставити лише ICANN (Інтернет корпорація з присвоєння імен та номерів), яка діє в інтересах всього інтернет-товариства з дотриманням міжнародних конвенцій та місцевих законодавств. Домен може відрізнятись за належністю до територіальних зон, або типів організації, наприклад: .ua, .us, .fr, .com, .edu, .net. Домен другого рівня – це унікальна назва сайту. Наприклад google.com. Домен третього рівня зазвичай використовується для створення невеликих сайтів, або підрозділів головного сайту, наприклад, форум. Як приклад домену третього рівня може бути test.site.ua [8].

За статистикою на сьогодні в мережі інтернет існує майже 2 мільярди вебсайтів, і їх число постійно збільшується. За даними моніторингової компанії [9] крива щорічного зростання кількості сайтів у світі має експоненційний тренд (рис. 1.1). Хоча дійсно активними є всього біля 200 млн.

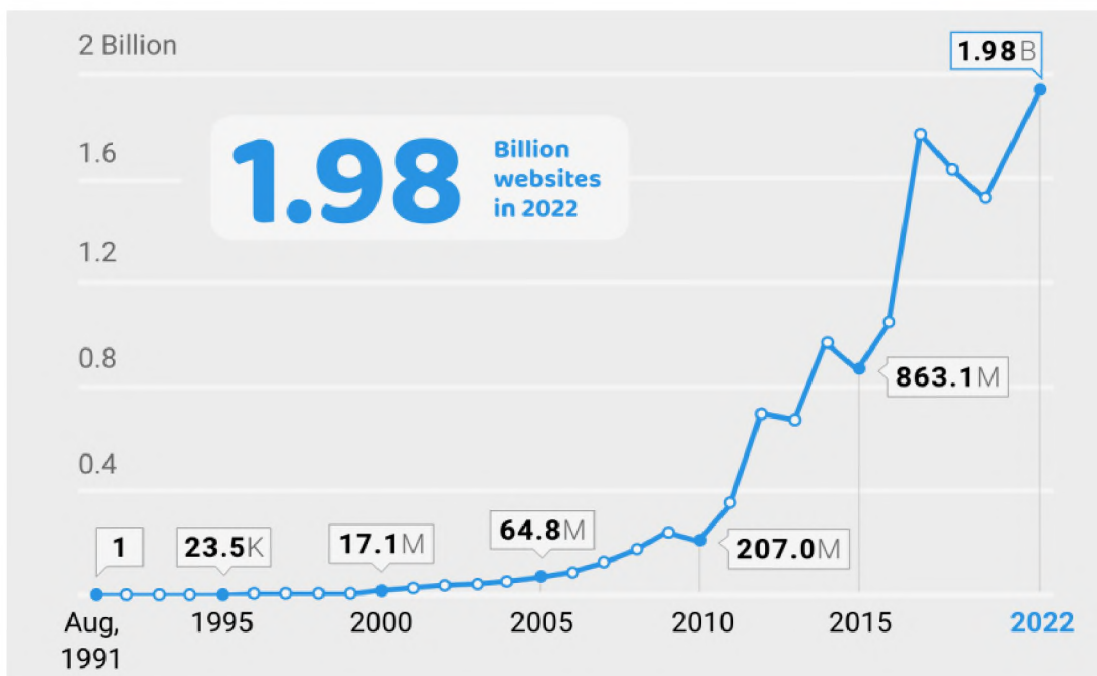


Рисунок 1.1 – Статистика зростання кількості вебсайтів у світі станом на початок 2023 р. (за даними [9])

Вебсайт – це ефективний інструмент маркетингу, який захоплює увагу аудиторії. Сайт повинен зацікавити користувача, а потім підштовхнути його до виконання якоїсь дії, наприклад до покупки товару.

Необхідність професійно створеного корпоративного сайту усвідомила велика кількість керівників. Більшість фірм вкладають кошти для надання своєму сайту сучасних функцій і ознак: це необхідний чинник для існування сайту, який дозволяє привертати додаткову аудиторію [8].

Це можуть бути як особисті сторінки, на яких викладена інформація про автора, так і інші види сайтів, серед яких:

- інформаційні сайти – сайти корпорацій, фірм і інше;
- сайти-візитки – основна інформація про певну, переважно невелику компанію або особу, контактні дані, перелік послуг;

- сайти-портали – великі вебсайти, на яких розташована велика кількість сервісів, такі як: поштові скриньки, форуми, каталоги зображень;
- інтернет-магазини – створені спеціально для популярної покупки будь-яких товарів через інтернет;
- навігаційні сайти – призначені для пошуку інформації та зручної навігації в інтернеті;
- соціальні сайти – різні соціальні мережі.

Слід зазначити, що близько 75% вебсайтів сьогодні не активні, однак є припаркованими доменами або подібними. Більшість відомих і популярних світових вебдодатків, мереж була розроблена вже до 2010 р. (табл. 1.1).

Таблиця 1.1 – Хронологія появи відомих світових вебзастосувань [10]

Рік	Кількість вебсайтів	Зміна за рік	Кількість користувачів інтернету	Користувачів на сайт, осіб	Назви вебдодатків, платформ, мереж, що з'явилися
2010	206 956 723	-13%	2 045 865 660	9.9	Pinterest, Instagram
2009	238 027 855	38%	1 766 206 240	7.4	
2008	172 338 726	41%	1 571 601 630	9.1	Dropbox
2007	121 892 559	43%	1 373 327 790	11.3	Tumblr
2006	85 507 314	32%	1 160 335 280	13.6	Twitter
2005	64 780 617	26%	1 027 580 990	16	YouTube, Reddit
2004	51 611 646	26%	910 060 180	18	The facebook, Flickr
2003	40 912 332	6%	778 555 680	19	WordPress, LinkedIn
2002	38 760 373	32%	662 663 600	17	
2001	29 254 370	71%	500 609 240	17	Wikipedia
2000	17 087 182	438%	413 425 190	24	Baidu
1999	3 177 453	32%	280 866 670	88	PayPal
1998	2 410 067	116%	188 023 930	78	Google
1997	1 117 255	334%	120 758 310	108	Yandex, Netflix
1996	257 601	996%	77 433 860	301	
1995	23 500	758%	44 838 900	1,908	Altavista, Amazon, AuctionWeb
1994	2 738	2006%	25 454 590	9,297	Yahoo
1993	130	1200%	14 161 570	108,935	
1991	1				World Wide Web Project

Як бачимо (див. табл. 1.1) всі потужні та корисні вебзастосування в області комерції, інформаційних послуг, без яких сьогодні неможливо уявити життя кожної людини, з'явилися протягом 10-15 років від винайдення ідеї гіпертексту і появи першого вебсайту. Це свідчить про надзвичайно швидкі темпи розвитку самих вебтехнологій та потужностей мереж, високий рівень залученості кожного користувача до інтернет-технологій.

На сучасному етапі сайт бізнес-компанії являє собою інтернет-портал, який забезпечує відвідувачу різну інформації про організацію, її діяльність, перелік послуг і т. д. Окрім цього, такий сайт може мати інформацію про виконані роботи, партнерів і відповіді на питання, які частіше за все задають користувачі [11].

Основна ціль будь-якого сайту бізнес-компанії – обслуговування користувачів таким чином, щоб це приносило вигоду і самій організації, і задовольняло потреби клієнтів у певних товарах, послугах, інформації тощо. Кілька прикладів вебсайтів свідчать про їхню важливість для будь-якого виду бізнесу, наприклад, з монтажу систем вентиляції (рис. 1.2).

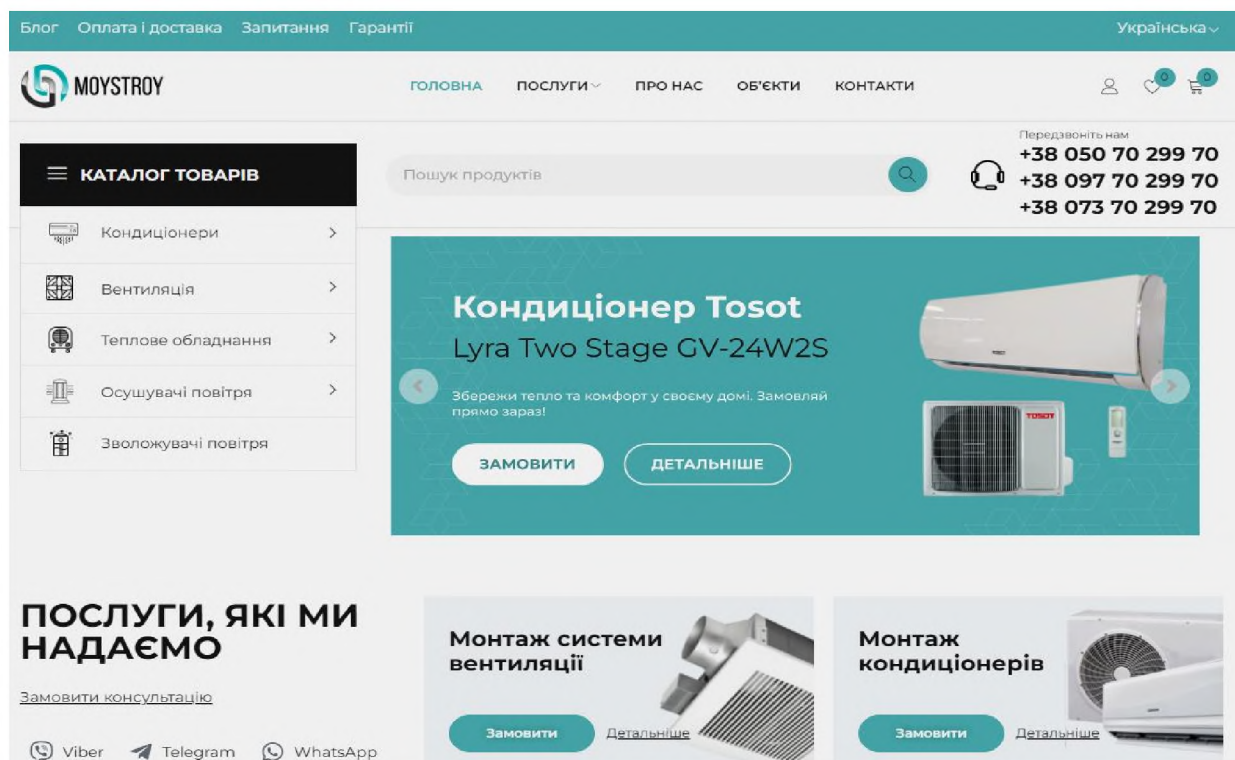


Рисунок 1.2 – Приклад оформлення промо-сайту компанії Moystroy [12]

Вебсайт (див. рис. 1.2) за змістом та дизайном відповідає місії та профілю діяльності компанії – представлені перелік послуг, змістовне меню, чітка графіка та мінімалістичний дизайн. На внутрішніх сторінках – структурована інформація для клієнтів, діловий стиль, що все разом викликає довіру до організації та професійності послуг.

Вебсайт компанії з надання іншого виду послуг – маркетингу та інтернет-реклами [13], показано на рис. 1.3.

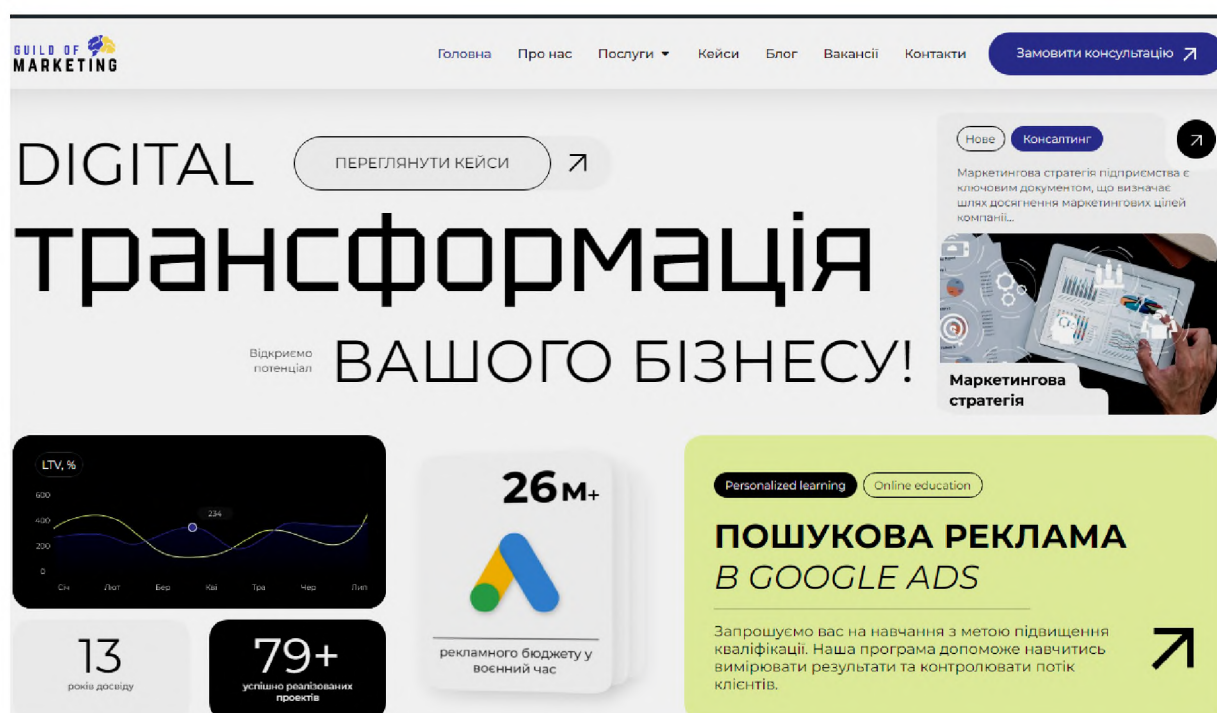


Рисунок 1.3 – Приклад оформлення та меню сайту компанії Leosvit Marketing

На промо-сайті (див. рис. 1.2 - 1.3) найчастіше розташовано необхідний перелік розділів: «Про нас», «Послуги», «Ціни», «Портфоліо» або кейси та інші. На кожній сторінці максимально необхідна й чітко висловлена інформація, графіка доречна, мінімум кольорів та вдале поєднання.

Промо-сайти розробляють, щоб акцентувати увагу на певному виді продукції, торгової марки або послуг. Такі веброзробки дозволяють залучити більше користувачів, створити клієнтську мережу відрізняються в першу чергу обсягом: як правило, це сайти з більш ніж 10-ма сторінками або розділами. Такі ресурси зазвичай мають значну кількість змістовного контенту. Дизайн

інформаційних сайтів простий, без зайвих прикрас. Головне – правильна подача контенту [14]. На такому вебсайті може рекламуватися і ціла компанія, і окремий продукт, вагома подія (конференція, фестиваль, конкурс тощо). Ідеальним рішенням є розроблення промо-сайту як для підтримки системного бізнесу, так і при виведенні нового продукту. Тому в центрі уваги є дизайн, наявність особливих прийомів із додаванням інфографіки, інтерактивності, реалізація через оформлення певної ідеї, та ін.

Отже, кожному підприємству, бізнес-компанії, організації об'єктивно необхідно мати свій власний, привабливий і добре структурований промо-сайт.

1.2 Огляд базових технологій для розроблення вебсайтів

Процес створення вебсайтів відбувається в декілька етапів та включає такі елементи:

- попереднє обговорення та збір інформації про ринок і компанію;
- розробка і затвердження технічного завдання на створення сайту;
- розробка структурної схеми сайту – положення елементів на сторінці;
- розробка дизайну сайту;
- розробка програмного коду, баз даних та інших елементів необхідних для функціонування сайту;
- тестування і розміщення сайту в інтернеті.

Структура – це логіка побудови сайту, ієрархічна організація взаємозв'язків сторінок. Продумувати структуру сайту потрібно ще до запуску проєкту, оскільки без чіткої логіки ми отримаємо просто набір документів Її простіше всього уявити у графічному вигляді – блок схема або майнд-карта, на якій чітко видно зв'язки, рівні вкладеності та ієрархія сторінок [14].

У кожній сторінки є своя унікальна адреса, в якій прописаний шлях до неї від головної сторінки, що відображається в її URL. Якщо структура сайту продумана грамотно, між сторінками встановлені гіперпосилання, то до будь-

якої з них у користувача буде доступ. Дуже важливо, щоб користувач швидко розібрався, як влаштований сайт і як йому потрапити на необхідну сторінку [14]. Для цього на етапі планування створюється схема внутрішньої структури майбутнього сайту (рис. 1.4).

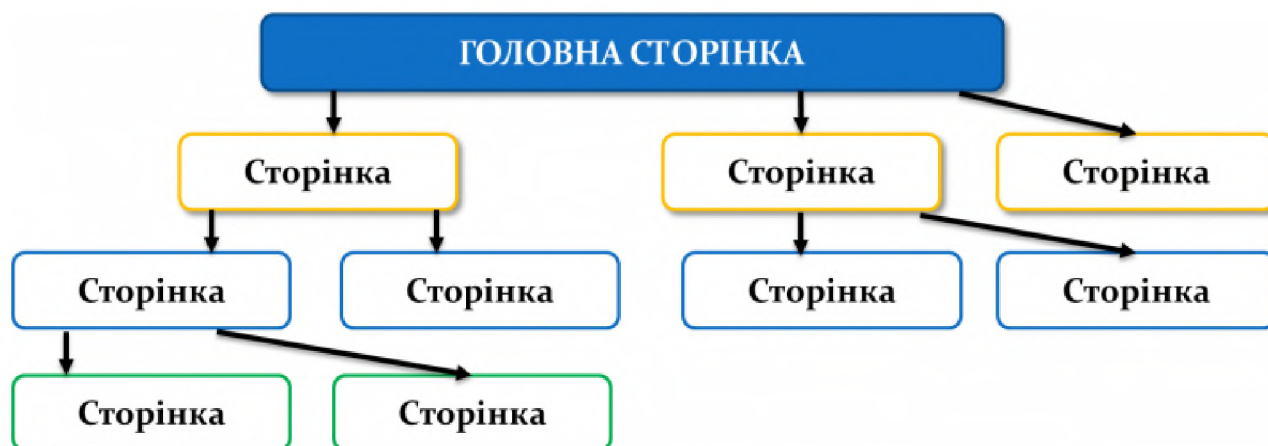


Рисунок 1.4 – Приклад схеми внутрішньої структури сайту

Структура макету окремих сторінок вебсайту в більшості випадків складається з таких основних елементів:

- шапка у верхній частині сторінки, де розташовується логотип, клік по якому повертає на Головну сторінку;
- бокове меню. Може бути як справа, так і зліва. На ньому розміщують елементи навігації і допоміжну інформацію;
- основна частина сайту. Те, заради чого користувач заходить на сайт. Основна інформація повинна бути читабельною і розміщуватись на першому-другому екрані;
- нижня частина сторінки. Зазвичай там знаходиться контакти, кнопки соціальних мереж, додаткова інформація [15].

Після створення структури сайту можна переходити до розробки зовнішнього вигляду сайту (дизайну інтерфейсу).

Найзручніше зовнішній вигляд сайту представити в графічному редакторі. Серед найбільш популярних рішень є: Photoshop, Figma, Sketch, Zeplin. Після цього готовий дизайн, який по суті є зображенням майбутнього

сайту, відтворюється у вигляді базової HTML-сторінки з подальшим застосуванням інших технологій.

В основі так званої «ручної» верстки вебсайтів лежить HTML – мова розмітки документів для відображення їх у браузері. Загалом, все що стосується зовнішнього виду сайту на боці клієнта відноситься до частини, яка називається frontend. На мові HTML автори описують структуру сторінок за допомогою розмітки. Елементи мови позначають частини вмісту, такі як параграф `<p>`, список `<ol>`, `<ul>`, `<li>`, таблиця `<table>` тощо.

На практиці при роботі над зовнішнім виглядом сайту використовуються додаткові добре розвинуті за останнє десятиріччя технології, такі як: CSS (Cascading Style Sheets) – мова опису зовнішнього вигляду документу, JS (JavaScript) – мова сценаріїв для придання інтерактивності вебсторінкам. JS використовується майже всіма сучасними сайтами в інтернеті [16].

Разом із графікою та сценаріями HTML і CSS є основою створення вебсторінок і вебдодатків. Актуальною версією CSS на даний момент є CSS3, більш нова CSS4 іще знаходиться у розробці. По суті, таблиця стилів – це спеціальний розділ в головній частині сайту в тегах `<style>...</style>` або файл, в якому описується, як буде виглядати кожен із елементів на сайті [17]. Завдяки цьому в HTML-документі залишається лише структура сторінки: блоки, їх вміст, розташування. Наприклад, на рис. 1.5 показано, як описати стиль для встановлення розміру шрифту заголовка в умовних одиницях.

```
<HTML>
  <STYLE>
    H1 { font-size: 2em }
  </STYLE>
  <BODY>
    <H1>Movies</H1>
  </BODY>
</HTML>
```

Рисунок 1.5 – Приклад запису властивостей заголовка в таблиці стилів CSS

Переваг використання стилів чимало, зокрема досягнення єдиного стильового оформлення однакових елементів, економія та раціоналізація коду,

швидкість завантаження сторінки та інші. CSS – мова для опису представлення вебсторінок, включаючи кольори, макет і шрифти. Це дозволяє адаптувати презентацію до різних типів пристроїв, таких як великі екрани, маленькі екрани або принтери. CSS не залежить від HTML і може використовуватися з будь-якою мовою розмітки на основі XML. Відокремлення HTML від CSS полегшує підтримку сайтів, спільний доступ до таблиць стилів між сторінками та адаптацію сторінок до різних середовищ.

Існують також CSS-фреймворки, які ще більш спрощують роботу верстальника, серед яких найпоширеніші: Bootstrap, Foundation, Tailwind. А також CSS-препроцесори – надбудови над CSS, які дозволяють писати спрощені синтаксичні конструкції, що підвищує простоту написання і редагування коду. Серед популярних препроцесорів можна виділити: Less, Sass (SCSS), Stylus [18].

Мова програмування JavaScript використовується програмістами для створення динамічних інтерактивних вебсторінок. JS настільки популярна, що її використання на сайтах сягає майже 98% (рис. 1.6).

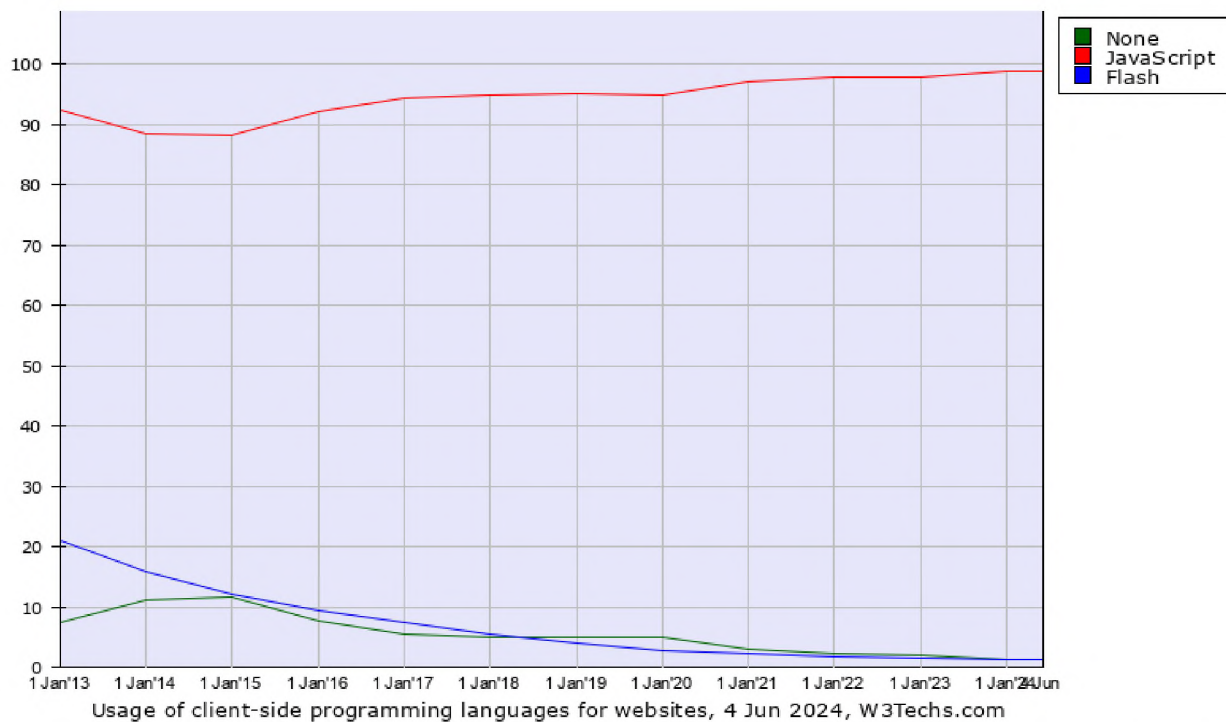


Рисунок 1.6 – Звіт W3Techs про використання JavaScript 2013-2023 pp. [19]

Графік (див. рис. 1.6) демонструє статистику використання JavaScript як мови програмування на стороні клієнта в інтернеті станом на початок 2024 р. З появою JS розробники змогли впроваджувати такий функціонал: ефекти при наведенні на елемент, показ та приховування елементів на сторінці, створення галерей в форматі каруселі, відтворення аудіо та відео і багато інших. Аналогами JavaScript є мови програмування: Dart, ASP.NET, CoffeeScript.

При розробці сучасних сайтів, «ванільний» JS майже не використовують, замість цього використовують фреймворки і бібліотеки. JavaScript-бібліотека – це колекція JS коду, який надає веброзробнику готові рішення для повсякденних задач. JavaScript-фреймворк також надає готові рішення, але також накладає певний стиль програмування [20].

Фреймворки (табл. 1.2) являють собою чітку структуру додатку і реалізуються з використанням так званих патернів проєктування. Найбільш широко використовувані патерни це: MVC (Model-View-Controller), MVP (Model-View-Persistor), MVVW (Model-View-ViewModel). Додатків на фреймворках доволі багато, і цей сегмент тільки набирає темп [21].

Таблиця 1.2 – Порівняння фреймворків мови програмування JavaScript

Назва	Переваги	Недоліки
React	Ефективність, гнучкість, зручна система багаторазових компонентів, підтримка спільноти, просте налагодження	Погана документація, важкість пошукової оптимізації
Angular	Великий вибір функцій, ймовірність збоїв дуже мала, отримання даних напряму, підтримка Google та потужна екосистема	Високий поріг входу, важкість міграції за старої версії на нову
Vue.js	Простота, легкість інтеграції, дружній інтерфейс, хороша документація	Відсутність підтримки від великих проєктів, відсутність плагінів
jQuery	Простота, прискорення роботи веброзробника, оптимізація під застарілі браузери, безліч готових рішення	Застарілий API, нема рівня даних, не підходить для комплексних задач

Найпопулярнішими фреймворками та бібліотеками є: React, Angular, Vue.js, jQuery. Ступінь їхньої популярності можна уявити за допомогою діаграми додатку А. Згідно аналітичних звітів компанії W3Tech [22], 17,8%

вебсайтів не використовують жодну з бібліотек JavaScript, які потрапили в зону опитування, jQuery використовується на 77,7 % усіх вебсайтів (верхня смуга), що складає 94,6 % частки ринку бібліотек JavaScript (нижня смуга діаграми).

Технології, що описують і відповідають за функціонал вебсайту, об'єднують в частину backend-розробки. До них відносять: HTTP-запити на сервер для отримання інформації з бази даних, реєстрація та авторизація користувача, додавання та видалення контенту на сайті і т. ін. [23]. Переваги і недоліки різних популярних мов програмування, що використовують для backend частини наведено в табл 1.3.

Таблиця 1.3 – Порівняльний аналіз мов програмування для backend

Назва	Переваги	Недоліки
PHP	Потужність і гнучкість, простота, велике ком'юніті, багато готових рішень	Менша захищеність порівняно з іншими мовами, неможливість створення десктопного застосунку
Ruby	Можливість використання лише необхідних бібліотек, швидкість розробки, надійний захист даних	Проблеми при роботі на Windows, мала кількість вакансій, використання великої кількості пам'яті
Python	Дуже легкий у вивченні, універсальність, дуже популярний, великий вибір готових рішень.	Повільність виконання, динамічна типізація, не сама зручна мова для мобільної розробки, місцями незручний синтаксис
Node.js	Швидкість виконання, кросплатформність, велика кількість інструментів	Важко писати великі проекти, залежність від бібліотек
Java	Незалежність від платформи, запуск в «пісочниці», автоматичне управління пам'яттю застосунка	Низька швидкість, важкість написання коду, плата за комерційне використання

JavaScript можна використовувати, наприклад, для створення форми замовлення компонентів, перевірки достовірності даних, введених користувачем, та передачі даних на сервер у проєкті вебсайту зі збирання ПК. За допомогою заданих функцій JavaScript можна проводити обрахунки даних та виводити результати на екран, не звертаючись до сервера. Крім того, JavaScript дозволяє взаємодіяти з API, що дає змогу отримувати найсвіжішу інформацію про компоненти безпосередньо з вебсайтів виробників. Ще однією важливою особливістю JavaScript є можливість додавання на сайт динамічних

елементів, таких як випадаючі меню, вкладки, повзунки тощо. Це підвищить естетичну привабливість сайту та покращить взаємодію з користувачами.

Анімація, зокрема анімовані графіки, діаграми, графіки тощо, може бути створена за допомогою JavaScript. Це може допомогти у візуалізації даних і підвищити привабливість вебсайту для користувачів.

1.3 Поняття UI/UX дизайну вебсайтів у розрізі сучасних трендів

Дизайн вебсайтів загалом прийнято розглядати у декількох площинах, зокрема, при виборі зовнішнього вигляду зручності навігації та представлення інформації, зручності архітектури зупиняються на поняттях UI/UX дизайні.

UI – User Interface, перекладається як «інтерфейс користувача». Інтерфейс являє собою графічну структуру програми. Це все, що бачить користувач на сайті: кнопки, тексти, зображення, поля введення тексту та всі інші елементи з якими може взаємодіяти користувач.

UX – User Experience, перекладається як «користувацький досвід». Визначається тим, як користувач взаємодіє з сервісом. Чи є структура сайту інтуїтивно зрозумілою, чи може навпаки, заплутує його? Досвід користувача визначається тим, наскільки йому легко чи складно взаємодіяти з елементами, які для цього призначені.

Дуже важливо щоб UI/UX були бездоганно виконані та ідеально узгоджені, адже від цього залежить те, чи сподобається користувачу сайт, чи захоче він скористатись послугами які є на цьому сайті, чи захоче повернутись на нього [24].

Від того, наскільки грамотно буде оформлена вебсторінка, буде залежати кількість користувачів і репутація організації. Важливо, щоб дизайн відображав напрямок діяльності компанії. Наприклад, якщо це сайт організації свят, то і їх домен повинен виглядати відповідним чином. А якщо це сайт банку, то використання яскравих кольорів і великих зображень буде

недоречним. Основними тенденціями дизайну вебсайтів, зокрема й сучасного промо-сайту є наступні:

- сторітеллінг: візуально оформлена історія, яка розповідається користувачу, мотивуючи його до подальших дій;
- великі простори: великий білий простір візуально збільшує екран, не дозволяє користувачу втратити концентрацію;
- адаптивний логотип: логотипи які в залежності від розміру екрану можуть підлаштовуватись під нього;
- адаптивність сайту: сайт повинен коректно відображатись на різних пристроях;
- яскраві кольори: кольори передають емоції, а емоції – найголовніше для користувача;
- повноекранне відео [25].

Тренди вебдизайну, так само як і звичайного дизайну, постійно змінюються. У 2023 році популярними стилями є:

1. Класичний стиль. Простий, лаконічний, акуратний. В такому дизайні всі елементи розташовані в очікуваних місцях, макет має зрозумілу навігацію і чітку структуру. Обов'язкова наявність назви і логотипу компанії, а також її слогану. Користується популярністю у адвокатських контор, юридичних фірм, сайтів промислових організацій[26].

2. Мінімалізм. За останні кілька років мінімалізм став головним трендом вебдизайну. До принципів мінімалізму належать – відсутність зайвих деталей, акцент в деталях, велика кількість вільного простору. Кольори використовуються прості і в невеликій кількості. Мінімалізм чудово підходить для сайтів таких напрямків як: IT, Travel, Fashion, особисті портфоліо [27] (рис. 1.7). Існує безліч абсолютно різних компаній, які використовують саме такі мінімалістичні вебсайти. Переваги цього стилю:

- висока швидкість завантаження сторінок через відсутність перевантаження графікою;
- правильна розстановка акцентів для виділення головного продукту;

– проста і зрозуміла структура, акуратний витончений дизайн.

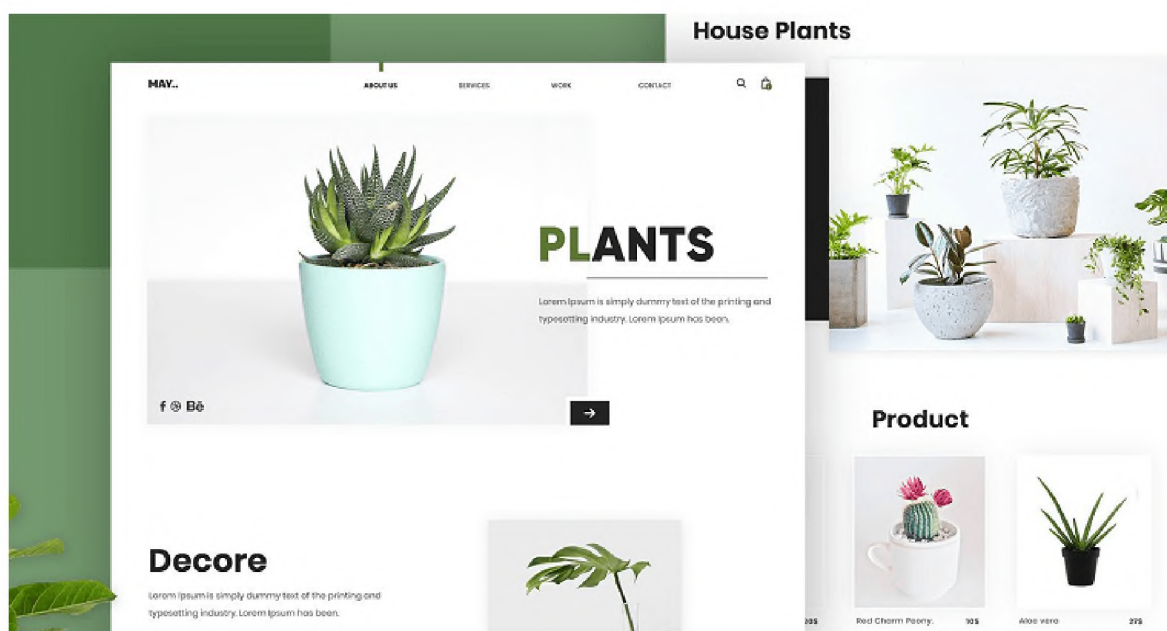


Рисунок 1.7 – Фрагмент вебсторінки з дизайном у мінімалістичному стилі

3. Метро стиль, або за іншими версіями ModernUI. Цей стиль був вперше представлений компанією Microsoft у 2010 році при дизайні власних продуктів. В основу даного стилю закладена орієнтація на типографіку, акцент на текст, плавна анімація і характерні плоскі прямокутні «плитки», інтерактивні картки і адаптивність. Завдяки цим елементам досягається доступність, інтуїтивне розуміння функціоналу, динамічність [28] (рис. 1.8).

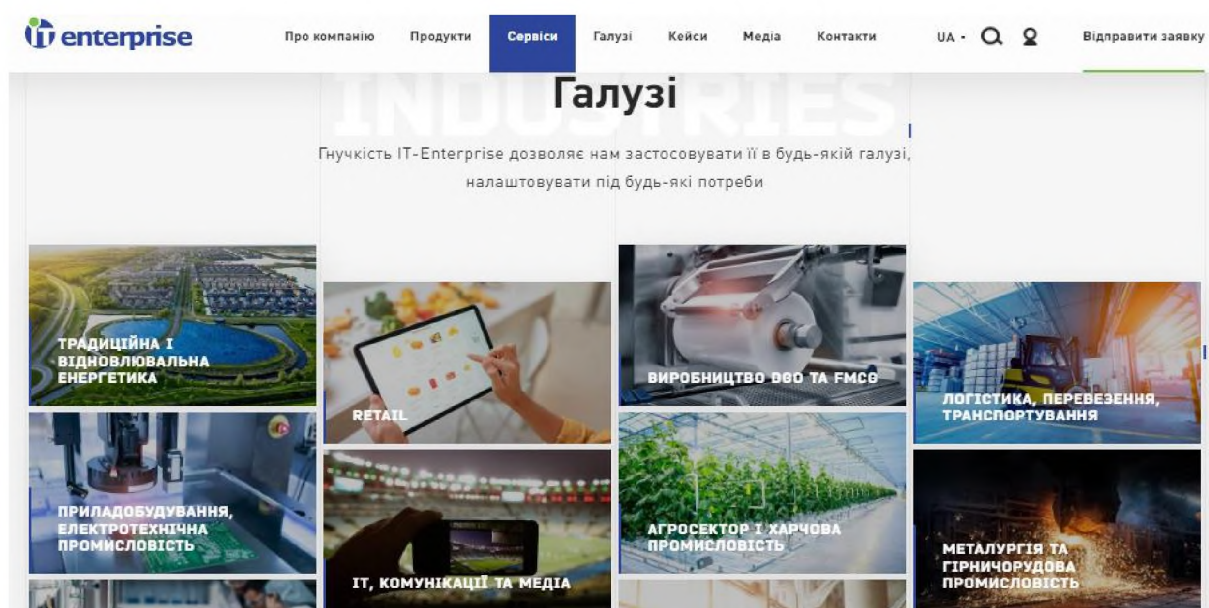


Рисунок 1.8 – Приклад дизайну сайту, виконаного в стилі метро

4. Окремо від стилю метро виділяють так зазваний плоский дизайн, або Flat Design, з характерним мінімалізмом і двовимірністю графіки. Flat або плоский стиль вперше з'явився у 90-х рр., а відновився з подачі Microsoft у 2010 р., коли прийшов на зміну пануючому скевоморфізму з його реалістичністю і об'ємами. Цей стиль виділяється застосуванням цікавої та незвичайної типографіки, великих зображень і відео, яскравих кольорів, дозволяє створювати унікальні проєкти [29] (рис. 1.9).

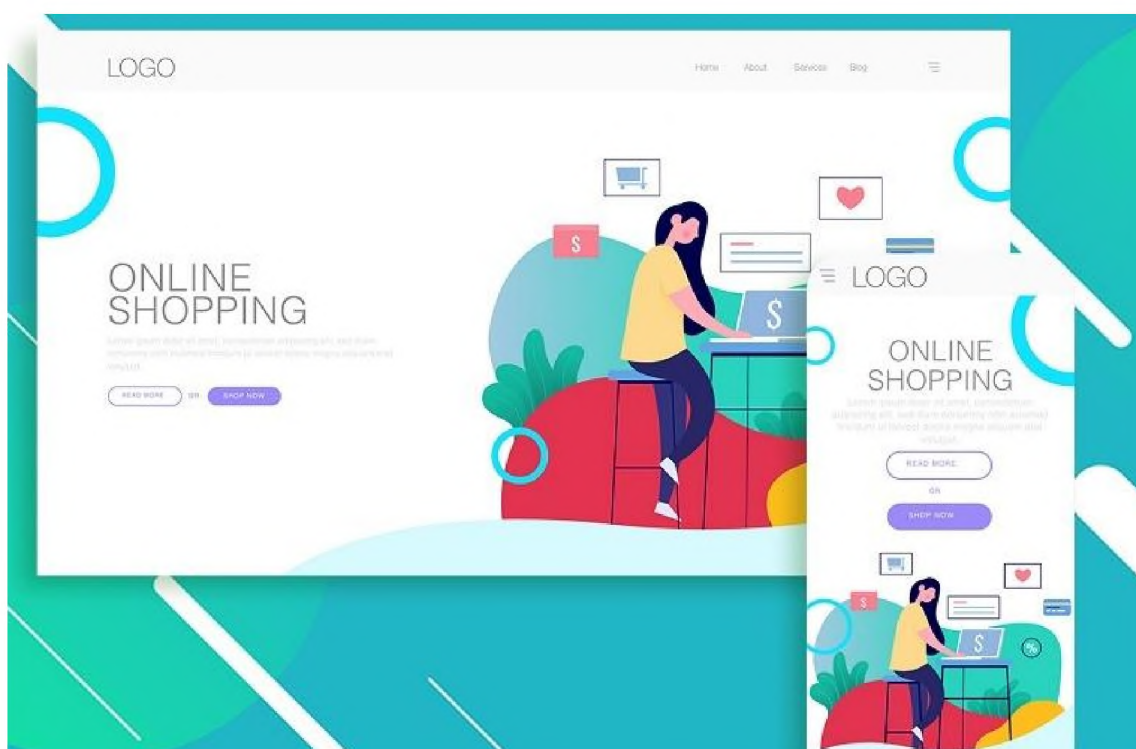


Рисунок 1.9 – Приклад дизайну сайту, виконаного в стилі Flat Design

Як бачимо (див. рис. 1.9), сайт відрізняє гарний підбір яскравих кольорів, плоскі іконки та великі плоскі кнопки. Загалом виглядає мінімалістично, але інформативно, викликаючи відчуття надійного та доступного сервісу. Чиста та акуратна сторінка дозволяє користувачеві легше зрозуміти зміст сторінки, не відволікаючи зайвими візуальними ефектами.

Отже, існує достатній вибір надійного інструментарію для розроблення промо-сайту для бізнес-компаній будь-якого спрямування, методи досягнення сучасного дизайну, які допомагають реалізувати кожну ідею.

РОЗДІЛ 2

ПЛАНУВАННЯ СТРУКТУРИ ТА МЕТОДОЛОГІЇ ПРОЄКТУВАННЯ ВЕБСАЙТУ ДЛЯ БІЗНЕСУ В СФЕРІ ІТ

2.1 Побудова концепції розроблення промо-сайту на тему комплектуючих та конфігурації комп'ютерів

Наявність власного вебсайту стає все більш важливою для різних груп користувачів, включаючи компанії, підприємців, творчих людей, брендів, оскільки люди все більше покладаються на інформацію й можливості, отримані в інтернеті. Промо-сайт відіграє вирішальну роль у взаємодії між компаніями та їхніми клієнтами у сфері різних товарів або послуг. Для більш предметного дослідження можливостей різноманітних технологій побудови сучасного за функціями і дизайном промо-сайту вирішено розробити в якості портфоліо вебсайт для бізнесу в сфері ІТ, предметною областю якого є збирання сучасних ПК та надання супровідних послуг. Така тематика недостатньо представлена в мережі у порівнянні з кількістю вебсайтів, на яких можна купити готовий ПК із усередненими характеристиками. Приклад такого сайту наведено на рис. 2.1.

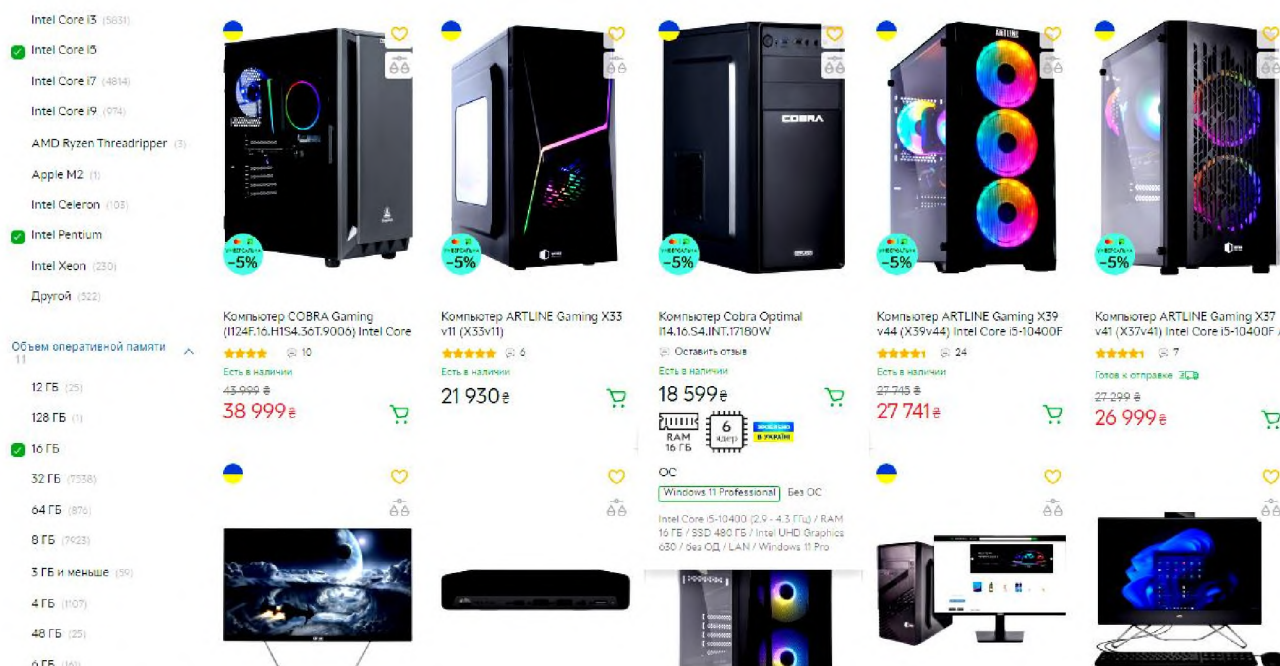


Рисунок. 2.1 – Приклад пропозиції на вебсайті з продажу ПК

За задумом, вебсайт для збирання ПК можуть запропонувати конфігуратори ПК, які дозволяють зібрати комп'ютер онлайн із перевіркою сумісності комплектуючих. Після проведення моніторингу послуг щодо варіантів придбання ПК, було вирішено розробити такий вебсайт, щоб користувач мав можливість змінити комплектацію представлених на сайті збірок, дізнатися ціну онлайн та порівняти характеристики. Після підбору комплектації оформити замовлення та отримати готовий ПК з абсолютно безкоштовною професійною збіркою. Такий вебсайт має бути розрахованим і для досвідчених користувачів, які знайдуть опис всіх характеристик для задоволення попиту, і для новачків, які тільки-но долучаються до користування ПК із абсолютно різними напрямками: ігри, дизайн, творчі проєкти, програмування, навчання та інші завдання.

З метою розширення функціональності сайту та більшої привабливості для відвідувачів можливо включити додаткові послуги, такі як встановлення програм, стрес-тест комп'ютерів та доставка готових ПК, навіть безкоштовне консультування та гарантійне обслуговування протягом певного часу.

Таким чином, уточненими завданнями для розроблення промо-сайту визначено такі:

1. Вебсайт стосується сфери бізнесу ІТ-компанії, яка надає послуги зі збирання ПК з комплектуючих, які пропонує ця фірма. Комплектуючі мають бути представлені на вебсайті з описом параметрів, ціною та іншими даними.

2. Готовий вебдодаток має виконувати функції конфігуратора ПК, тобто онлайн-інструмента, який дозволяє користувачам зібрати комп'ютер онлайн із перевіркою сумісності комплектуючих. Для збірки пропонується вибір компонентів, таких як процесор, охолодження, материнська плата, оперативна пам'ять, відеокарта та інші. Ціна розраховується автоматично в онлайн-режимі, послідовно формуючи вартість готового ПК.

3. Інтерфейс промо-сайту зі збирання ПК повинен бути інтуїтивно зрозумілий і стати помічником для пересічного користувача зібрати і купити системний блок з нуля.

4. На вебсайті варто передбачити форму зворотного зв'язку, щоб включити опцію про думку експерта. Мають бути вбудовані й інші форми комунікації: сторінки в соціальних мережах, месенджерах, або чат-бот.

5. Промо-сайт, як зазначено в розділі 1, повинен мати якісне стильове оформлення (наприклад, обираємо стиль метро), зрозумілий інтерфейс, інформативну графіку й контент.

6. Готовий вебсайт також повинен відповідати критеріям технічної оптимізації, перевірка якої здійснюється перед хостингом.

Залежно від бажаної продуктивності, бюджету користувача та рівня його компетентності, існують різні методи побудови ПК. Розглянемо деякі з них детальніше для побудови алгоритму дій на вебсайті в майбутньому.

1. Створення комп'ютера з нуля. Цей підхід передбачає купівлю окремих частин, таких як процесор, материнська плата, оперативна пам'ять, жорсткий диск та інші, і складання їх разом, щоб сформувати повноцінний комп'ютер. Цей підхід є найбільш адаптивним і дозволяє вибрати найкращі деталі, виходячи з бажаної продуктивності та цінового діапазону.

2. Купівля готового комп'ютера. Цей підхід передбачає придбання готового комп'ютера від виробника. Для людей, які не хочуть мати справу зі складанням, це може бути непоганим варіантом, але він також може бути менш універсальним і дорожчим, ніж збірка ПК з нуля.

3. Оновлення застарілого ПК. Цей метод передбачає заміну або покращення певних частин існуючого ПК, таких як процесор, оперативна пам'ять, жорсткий диск та інші. Для тих, хто хоче підвищити продуктивність свого ПК без необхідності купувати новий комп'ютер, цей варіант може виявитися гідним варіантом.

4. Купівля ноутбука в лізинг. Цей підхід передбачає короткострокову оренду вже зібраного комп'ютера для тих, кому комп'ютер потрібен тимчасово і які не хочуть витратити багато грошей на його купівлю.

Для того, щоб досягти максимальної продуктивності та свободи у виборі компонентів, для нашого проєкту було обрано збирання комп'ютера з нуля.

2.2 Обґрунтування вибору методів та середовища створення сайту

Для розроблення вебсайтів на сьогодні існує широкий набір технологій, зокрема, різноманітних конструкторів, мов програмування, бібліотек та ін. Для створення проєкту були використані такі технології, як HTML [30], CSS [31] та мова програмування JavaScript. Крім того, при проєктуванні сайту для створення графічного макету використано спеціальне програмне забезпечення, таке як Figma [32], а також Visual Studio Code [33] для написання коду. Створення проєкту передбачало наступні етапи: аналіз потреб користувачів, проєктування інтерфейсу, структури і макету сайту, написання коду і тестування.

Для створення сайтів, подібних до поточного проєкту, використовують спеціалізоване інтегроване середовище (IDE) Visual Studio Code [33]. Visual Studio Code це сучасне інтегроване середовище розробки, створене компанією Microsoft для розробки GUI (графічного інтерфейсу користувача), консолі, вебпрограм, мобільноорієнтованих додатків, хмарних і вебсервісів тощо. За допомогою цієї IDE можна створювати керований код, а також рідний код. Він використовує різні платформи програмного забезпечення Microsoft для розробки програмного забезпечення, такі, як магазин Windows, Microsoft Silverlight і Windows API тощо. Це IDE не призначене лише для певної мови, його можна використовувати для написання коду на мовах C#, C++, VB (Visual Basic), Python, JavaScript і багато інших мов.

Основна відмінність між IDE та редакторами JavaScript полягає в тому, що останній не має багатьох функцій налагодження. Крім того, IDE підтримують різні системи ALM, такі як Git або GitHub. Однак, оскільки більшість редакторів коду підтримують ці системи, у цьому відношенні різниця поступово згладжується. У багатьох відношеннях IDE для розробки JavaScript є чудовими. Вони прискорюють процес розробки та роблять його ефективнішим. Крім того, їх здатність миттєво тестувати речі приносить численні переваги. Однак, незважаючи на це, редактори коду все ще популярні,

оскільки багато з них ви можете отримати безкоштовно. Вони також легші, швидші та менш ресурсномісткі, тому їх можна використовувати на ноутбуках, смартфонах. Крім того, щоб зробити речі ще більш заплутаними, частина функціональних можливостей IDE вводиться в редактори коду. Це перетворює їх із розкішних блокнотів (вони були 10 років тому) на IDE-подібні платформи. Один із рейтингів IDE [34] для веброзробки наведено на рис. 2.2.

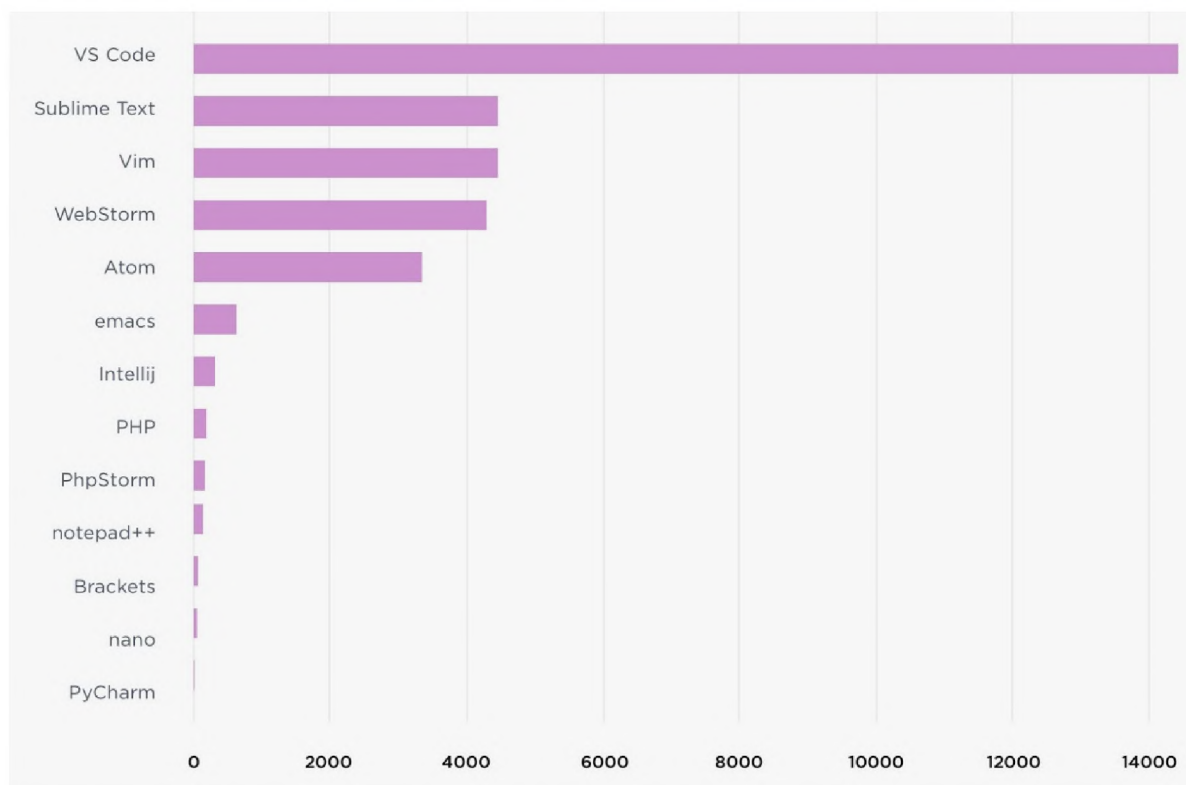


Рисунок 2.2 – Діаграма популярності редакторів коду та IDE для JS

За даними моніторингів (див. рис. 2.2) першість серед IDE для JS тримає середовище програмування і розробки Visual Studio Code. За ним йдуть SublimeText, Vim, WebStorm і Atom. Visual Studio Code (VS) має понад 14 тисяч користувачів із 20 тисяч учасників опитування [34]. Окрім JS, він підтримує 30 мов програмування, таких як Ruby, Python, C# тощо.

Підсвічування синтаксису, автозавершення, налагодження коду, взаємодія з системами контролю версій та багато інших важливих функцій доступні у Visual Studio Code. Ці функції допомагають прискорити розробку та підвищити якість коду.

Visual Studio Code може встановити велика кількість користувачів, оскільки це крос-платформний редактор, який працює на Windows, MacOS і Linux. Чому варто розробляти проєкти, схожі на вебсайти, за допомогою Visual Studio Code?

Visual Studio Code – це практичний та ефективний інструмент для розробки вебзастосунків. Завдяки широким можливостям, функціональності та зручному інтерфейсу процес розробки є більш ефективним та результативним. Крім того, Visual Studio Code має відкритий вихідний код і є безкоштовним, що робить його доступним для всіх користувачів. Початок інсталяції й вибору операційної системи для встановлення наведено на рис. 2.3.



Рисунок 2.3 – Вибір потрібної платформи на початку інсталяції VS Code

До Visual Studio Code також включено потужні можливості налагодження коду, що дозволяє швидко знаходити та усувати помилки. Завдяки вбудованому відладчику ви можете розміщувати точки зупинки у вашому коді та досліджувати кожну стадію виконання програми.

Крім того, Visual Studio Code пропонує потужну екосистему розширень, які можна використовувати для персоналізації та розширення можливостей редактора. Існують розширення для роботи з різними мовами програмування,

інтеграції з системами контролю версій, інструменти для підвищення продуктивності та багато іншого [35].

Використання Visual Studio Code для розробки вебзастосунків має кілька переваг: містить багато можливостей для налаштування та розширення функцій, є безкоштовним і з відкритим вихідним кодом, а також дозволяє легко створювати та редагувати код. Тому використання Visual Studio Code у нашому проєкті виявилось корисним і успішним. Це дозволило створити високоякісний код і успішно завершити проєкт.

Для створення інтерактивності на вебсторінці заплановано використовувати JavaScript, який є безумовним лідером серед мов програмування для веброзробки [36]. Популярність JavaScript така висока, тому що він має низку суттєвих переваг:

1. Швидкість. Клієнтська сторона JS дуже швидко працює, тому що він може працювати прямо в браузері, поки не потрібна інформація ззовні (запит на сервер), швидкодія JS не змінюється.
2. Простота. JS відносно легко вивчити і застосовувати.
3. Сумісність. JS добре поводить себе з іншими мовами програмування і може застосовуватись у великій кількості застосунків.

Рейтинг мов програмування за 2023 р. показує популярність JS (рис.2.4).



Рисунок 2.4 – Популярність мов програмування у 2023 р. (за даними [37])

До мінусів JS можна віднести те, що він не завжди коректно працює з різними браузерами, а також, оскільки JS виконується на стороні клієнта, то існує більша ймовірність злому клієнтських даних. Створений сайт необхідно буде перевірити на кросбраузерність, тобто пересвідчитись, що він відображається і працює коректно в популярних браузерах. Для цього використовуються браузери: Google Chrome, Mozilla Firefox, Opera, Safari.

2.3 Проєктування макету промо-сайту в середовищі Figma

Для створення дизайну макету вебсайту було прийнято рішення використовувати онлайн-сервіс для розробки інтерфейсів і макетів Figma. Figma – це сучасна онлайн-платформа для проєктування користувацьких інтерфейсів, яка пропонує різноманітні інструменти для роботи з макетами та дизайном, що дозволяє розробникам і дизайнерам створювати оригінальні та стильні сайти в найкоротші терміни [32]. Робота з рисунком-логотипом програми показано на рис. 2.5.

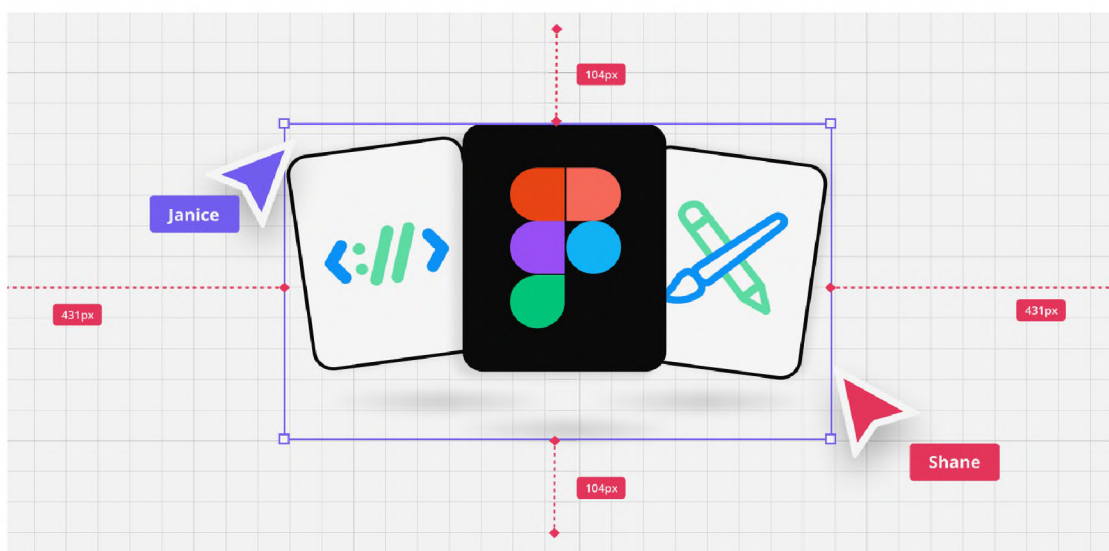


Рисунок 2.5 – Зображення логотипа та прикладу взаємодії в середовищі Figma

До суттєвих переваг Figma можна віднести те, що вся робота ведеться в хмарі, режим для розробників зручний, безкоштовність. Крім того, Figma

дозволяє працювати в команді в режимі реального часу, що полегшує співпрацю дизайнерів і розробників, а також дозволяє клієнтам вносити зміни і коментарі безпосередньо в макеті. Середовище Figma пропонує різноманітні інструменти для створення фігур, ліній і форм, а також для накладання тексту, вибору з декількох кольорових палітр, додавання піктограм і багато іншого. Всі ці інструменти дозволяють створювати високоточні та професійно виконані прототипи і макети вебсайтів.

Figma стала вирішальним інструментом для розробки макету та дизайну інтерфейсу сайту на початку виконання завдання з розроблення промо-сайту й вибору його тематичної спрямованості. За її допомоги було можна швидко вносити зміни та оновлення, а також розробити самобутній вебсайт. За допомогою Figma вдалося ретельно спланувати кожен аспект сайту, а також представити візуально готовий продукт ще до початку розробки. Фрагмент створення макету в Figma показано на рис. 2.6.



Рисунок 2.6 – Візуальне зображення макету дизайну сайту в середовищі Figma

Процес створення дизайну майбутнього вебсайту в Figma складається з наступних етапів:

1. Створення рамки проєкту – визначення обсягу робіт, вибір кольорової гами, шрифтів та інших елементів.

2. Розмітка – створення загальної структури сторінки та розміщення блоків.

3. Додавання елементів – додавання тексту, зображень, кнопок та інших елементів дизайну.

4. Взаємодія з елементами – додавання анімації та інтерактивності, щоб зробити дизайн більш динамічним та привабливим для користувачів.

5. Прототипування – створення прототипу, щоб показати, як буде виглядати та працювати сайт на різних екранах та пристроях.

6. Тестування – перевірка функціональності та коректності відображення дизайну на різних пристроях та екранах.

Повний дизайн розробленого макету в графічному середовищі представлено в додатку Б.

Результатом роботи, описаної в даному розділі є обрання методів та інструментарію для розробки сайту, створення календарного плану робіт, затвердження технічного завдання, створення схеми сайту, створення дизайну та макетів сторінок майбутнього сайту. Після цього можна приступати безпосередньо до програмної реалізації сайту, наповнення його контентом та тестування.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОЦЕСУ РОЗРОБЛЕННЯ ПРОМО-САЙТУ ДЛЯ БІЗНЕС-КОМПАНІЇ

3.1 Написання коду промо-сайту і формування проєкту у Visual Studio Code

Згідно з обґрунтуванням вибору (див. розділ 2) для написання коду та верстки були використані мови HTML, CSS та JS разом із середовищем розробки Visual Studio Code. Робота над програмною реалізацією сайту розпочинається із налаштування середовища розробки Visual Studio Code (рис. 3.1). Для цього потрібно створити папку проєкту, визначитись із типами і створити структуру файлів. Також потрібно підключити необхідні плагіни, такі як: автоформатування коду, підтримка синтаксису PHP, Emmet – для легшого написання HTML та CSS. Також для роботи с JavaScript необхідно встановити Live Server. Фрагмент робочої області на початку роботи наведено на рис. 3.1.

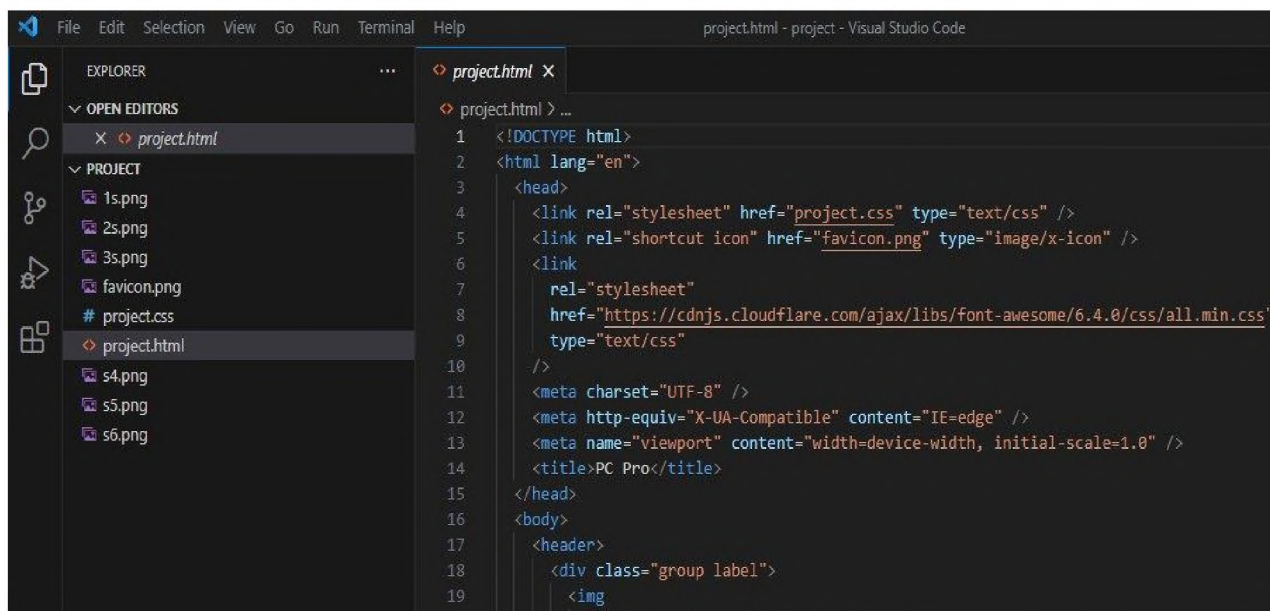


Рисунок 3.1 – Зміст HTML-файлу в середовищі VS Code і структура проєкту

Коли структура HTML-коду готова (додаток В), можна починати додавати стилі CSS, щоб зробити сайт більш привабливим і функціональним.

За допомогою CSS задається зовнішній вигляд і розташування компонентів на сторінці. На цьому кроці додаються стилі для різних елементів на сторінці, таких як шрифти, кольори, фони, рамки, трансформації тощо.

Написання коду HTML і CSS – це творчий процес, який вимагає розуміння того, які елементи потрібні на сторінці, і вміння працювати з різними властивостями CSS. Частина змісту файлу стилів CSS показано на рис. 3.2. Обидва файли розміщуються в папці Project на лівій панелі VS Code.

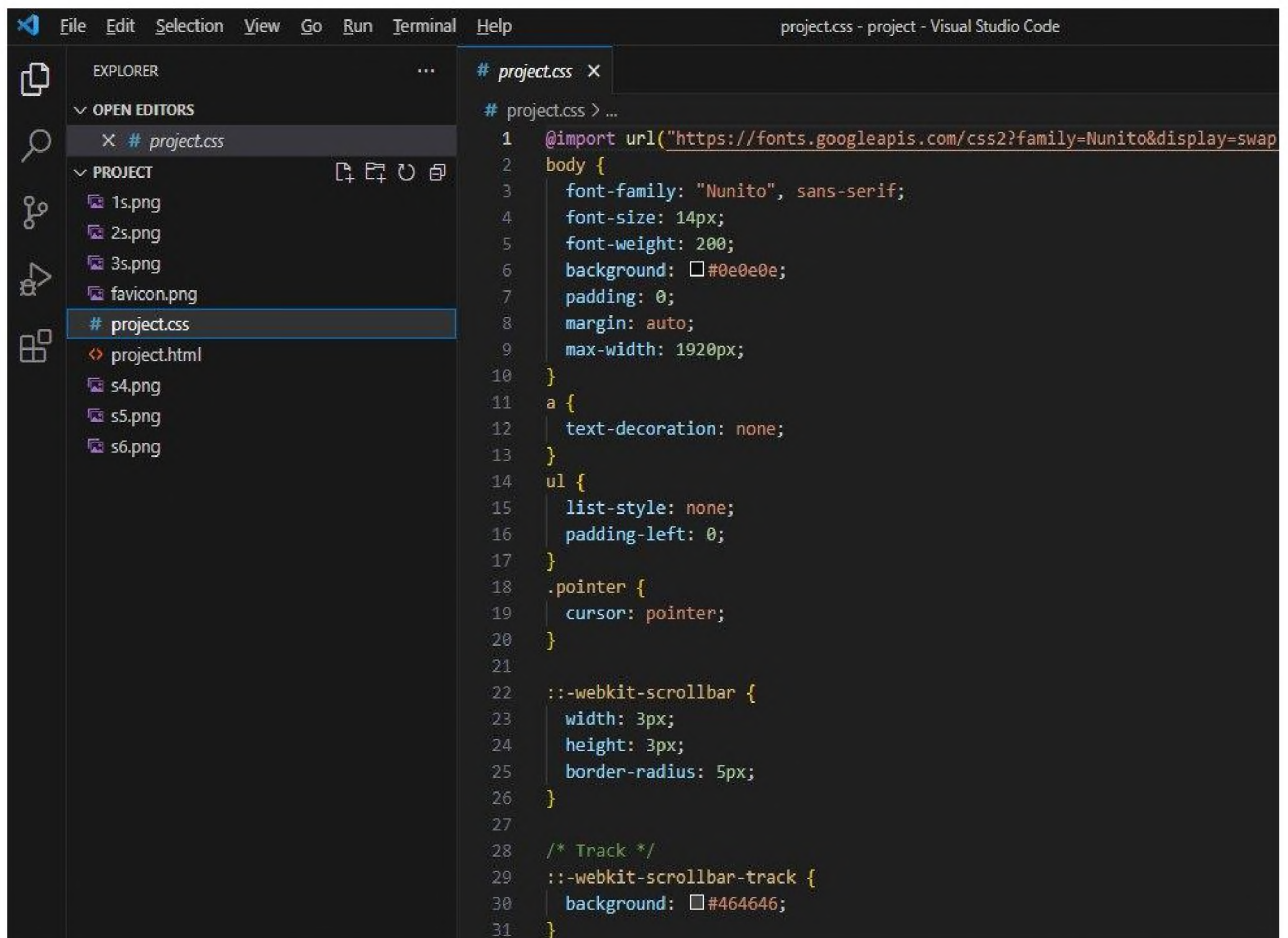


Рисунок 3.2 – Зображення фрагменту змісту CSS-файла у VS Code

Для створення інтерактивних елементів вебсторінок використовується JavaScript, який є важливим компонентом веброзробки. Використовуючи цю мову програмування, можна додавати динамічні компоненти на вебсайти, щоб вони реагували на введення користувача без необхідності оновлення сторінки.

Це особливо корисно для промо-сайтів, оскільки вони повинні мати інтерактивні компоненти для складних операцій, таких як візуалізація даних та

анімація. Кнопки, меню, форми, таблиці, графіки, карти та інші інтерактивні компоненти можуть бути створені на сторінках також за допомогою JavaScript. До того ж він дозволяє створювати інтерактивні графіки та діаграми на наукових вебсайтах, щоб відвідувачі могли досліджувати та оцінювати дані, змінювати налаштування та отримувати нові результати. Наприклад, запис функцій з обробки меню, де використані оператори циклу For, Return та оператори управління процесом If, Else, наведено на рис. 3.3. Повна версія всіх застосованих скриптів представлена в додатку Г.

```

343
344 let sub = document.querySelectorAll(".menu-content[data-menu-content] .submenu li");
345 for (let i = 0; i < sub.length; i++) {
346     if (sub[i] == target){
347         document
348             .querySelectorAll(".menu-content[data-menu-content].active .submenu li")
349             .forEach((e2) => {
350                 if (e2.dataset.category == target.dataset.category)
351                     e2.classList.add("active");
352                 else e2.classList.remove("active");
353             });
354         document.querySelector(
355             ".menu-content[data-menu-content].active .items"
356         ).innerHTML = buildCategory(
357             target.dataset.menu,
358             target.dataset.category
359         ).join("");
360         return;
361     }
362 }
363 let modalOpeners = document.querySelectorAll('[data-show-pc]');
364 for(var i = 0; i < modalOpeners.length; i++) {
365     if (modalOpeners[i] == target || target.closest('[data-show-pc]') == modalOpeners[i]){
366         console.log(target);
367         document.querySelector('.modal_container').innerHTML = buildPcBill(target.closest('d
368             `<button class="modal_close">&#10005;</button>`;
369         openModal();
370         return;
371     }
372 }
373 if (target.classList.contains('modal_close')) closeModal();
374 else if (target.classList.contains('filter')) {
375     document.querySelectorAll('.proposal li').forEach(prop =>{
376         if (![...selected.values()].every(sm=> pcList.get(parseInt(prop.dataset.showPc)).item
377             prop.remove();
378     })
379 }
  
```

Рисунок 3.3 – Зображення фрагменту скрипта JS при написанні в VS Code

Для розміщення елементів на сторінці використовується CSS Flexbox. Це технологія для створення гнучких макетів за рахунок правильного розміщення елементів на сторінці. За допомогою неї можна легко керувати дочірніми елементами, міняти їх напрям, відстані між ними та багато іншого.

Для спрощення створення адаптиву, а також для того щоб весь контент сайту не розпливався по краям, створюється css-клас `.container`, в якому задається максимальна ширина, зокрема, в нашому випадку 100 %. При роботі над проектом це значення можна буде змінювати в залежності від пристрою. Цей клас буде використовуватись для всіх майбутніх елементів сайту окрім фону хедера та футера. До властивостей класу додається `flex`.

У файлі `project.html` в тезі `<head>` встановлюється вся службова інформація, вказується титульний надпис вебсторінки. Зазвичай тут підключаються каскадні стилі (рис. 3.4).

```
<head>
  <link rel="stylesheet" href="project.css" type="text/css" />
  <link rel="shortcut icon" href="favicon.png" type="image/x-icon" />
  <link
    rel="stylesheet"
    href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css"
    type="text/css"
  />
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>PC Pro</title>
</head>
```

Рисунок 3.4 – Службова інформація головної вебсторінки в частині `<head>`

Далі, орієнтуючись на макет, починаємо створювати блоки з необхідним наповненням всередині тегу `<body>`, а стилі писати в файлі `project.css`. Повний опис стилів головної сторінки наведено в додатку Д.

Всі окремі елементи вебсайту зручно розміщувати в блокових `<div>` та інлайнових `<span>`. Ці теги не відображаються за замовченням, бо вони представляють собою контейнери, в які поміщається контент. Тому до них можна застосовувати безліч CSS-стилів, щоб досягнути бажаного зовнішнього вигляду.

Тег `<div>` використовується для групування елементів. В ньому можна розміщувати різного роду елементи, таких як: заголовки, абзаци, списки, інші `<div>` елементи.

Тег `<span>` використовується для виділення `inline` елементів, таких як окремі слова або речення.

Верстання відбувається зверху вниз, тому спершу створюємо хедер, який буде містити логотип сайту, а також коротке меню навігації. Цей елемент буде однаковий для всіх сторінок і буде відрізнятися лише виділеним елементом в меню навігації, в залежності від сторінки на якій ви знаходитесь в даний момент. Використано логотип компанії, підібрано google-шрифт, зокрема Nunito для дизайну текстових елементів (рис. 3.6).

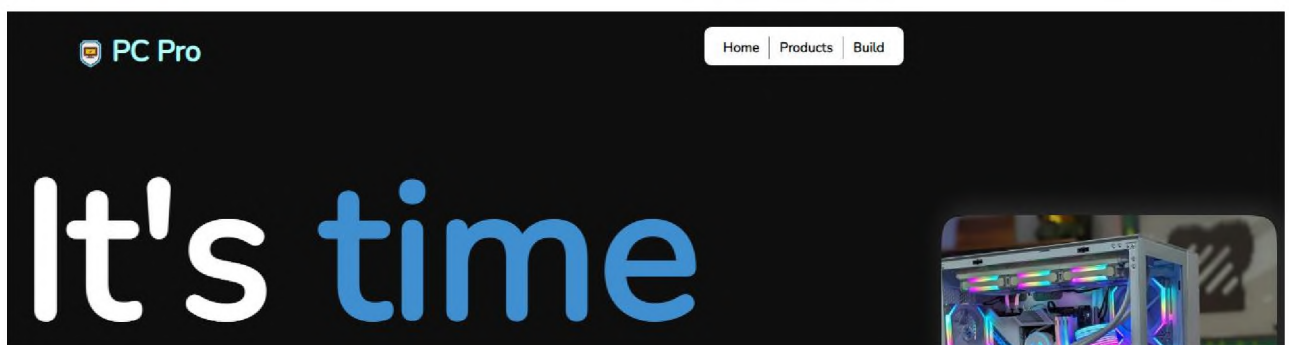


Рисунок 3.6 – Приклад дизайну верхньої частини сторінки (хедер) сайту із використанням google-шрифтів

Опис меню, наприклад, виконано як одна сегментована кнопка (рис. 3.7)

```

26 <div class="group buttons">
27   <a class="btn link" href="#">Home</a>
28   <a class="btn link" href="#selection">Products</a>
29   <a class="btn link" href="#picker">Build</a>
30 </div>
31 <div class="group right"></div>

```

Рисунок 3.7 – Приклад запису блоку меню у верхній частині (хедеру)

За властивості меню відповідає клас `header.group.buttons` у css-файлі з визначеним розміром, закругленими границями, фоном (рис. 3.8).

```

70 header .group.buttons {
71   display: flex;
72   flex-direction: row;
73   align-items: center;
74   padding: 10px;
75   background-color: white;
76   border-radius: 8px;
77 }
78 header .group.buttons .btn:not(:first-child) {
79   border-left: 1px solid gray;
80 }

```

Рисунок 3.8 – Запис властивостей для групи кнопок меню в CSS-файлі

Після створення макету та написання коду здійснено первинне тестування та налагодження сайту. Під час тестування були виявлені численні помилки та дефекти, які виправлені, щоб сайт працював належним чином і стабільно (рис. 3.9).

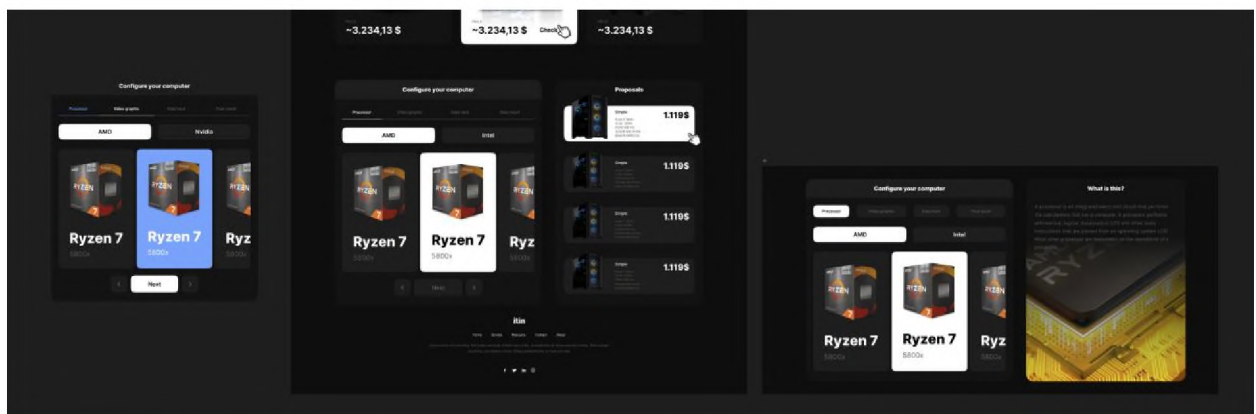


Рисунок 3.9 – Зображення окремих доповнень у Figma після етапу тестування

Зрозуміло, що розробка сайту зі збірки комп'ютерів включає в себе низку етапів, які виходять за рамки суто технічних і включають в себе обговорення та розробку дизайну, тестування та усунення несправностей. У результаті виконання роботи вдалося розробити вебсайт, який відповідав усім заданим специфікаціям і досягнув цілей проєкту, використовуючи такі інструменти та ресурси, як Figma та Visual Studio Code.

Опис інших технічних рішень, які були застосовані при розробленні й верстанні вебсайту доцільно провести з демонстрацією отриманих функцій сайту.

3.2 Опис процесу складання ПК як функції вебсайту

Для огляду функціональних характеристик готового сайту далі наведено перевірка й опис окремих дій користувача від початку до завершення комплектації комп'ютера за власним вибором складових. Як можна побачити на ілюстраціях (рис. 3.10 та рис. 3.11), створено окрему область на сторінці з назвами і фото частин комп'ютера, які користувач може вибрати самостійно поетапно від процесора, відеокарти та пам'яті, до готових конфігурацій з вибраних елементів. У процесі вибору компонентів є поля, в яких можна додати інформацію по кожному з комплектуючих, а також додати невелику ілюстрацію, як це буде виглядати в ПК і на що буде впливати в роботі з ним.

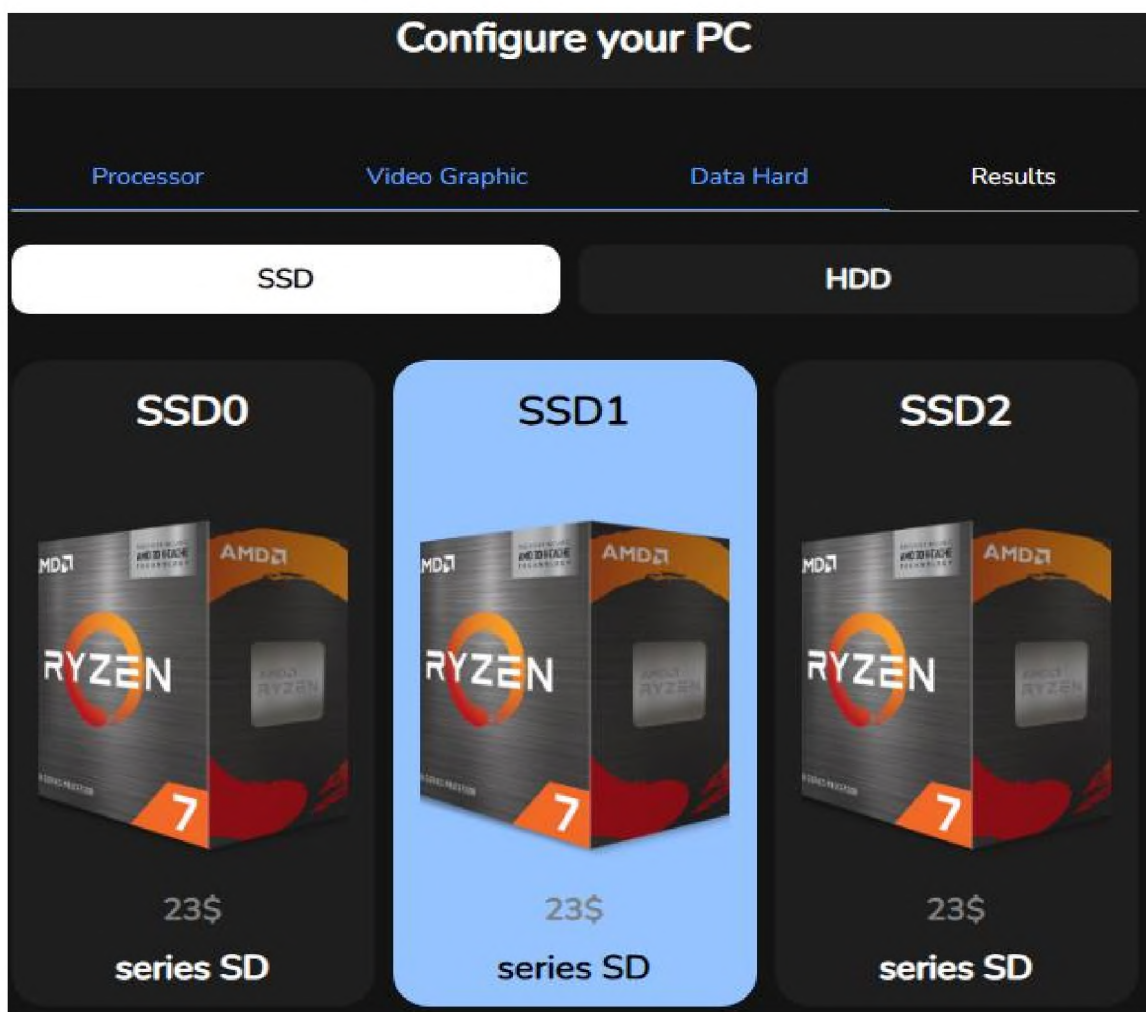


Рисунок 3.9 – Приклад подання опису комплектуючих (Data Hard, SSD), обраних для складу ПК на промо-сайті

Користувач може редагувати цей вибір, доповнювати його, звіряючись з тим, що бажає отримати в кінці. Після цього необхідно перейти до форми (вікно оформлення замовлення) за вибраними комплектуючими (рис. 3.10).

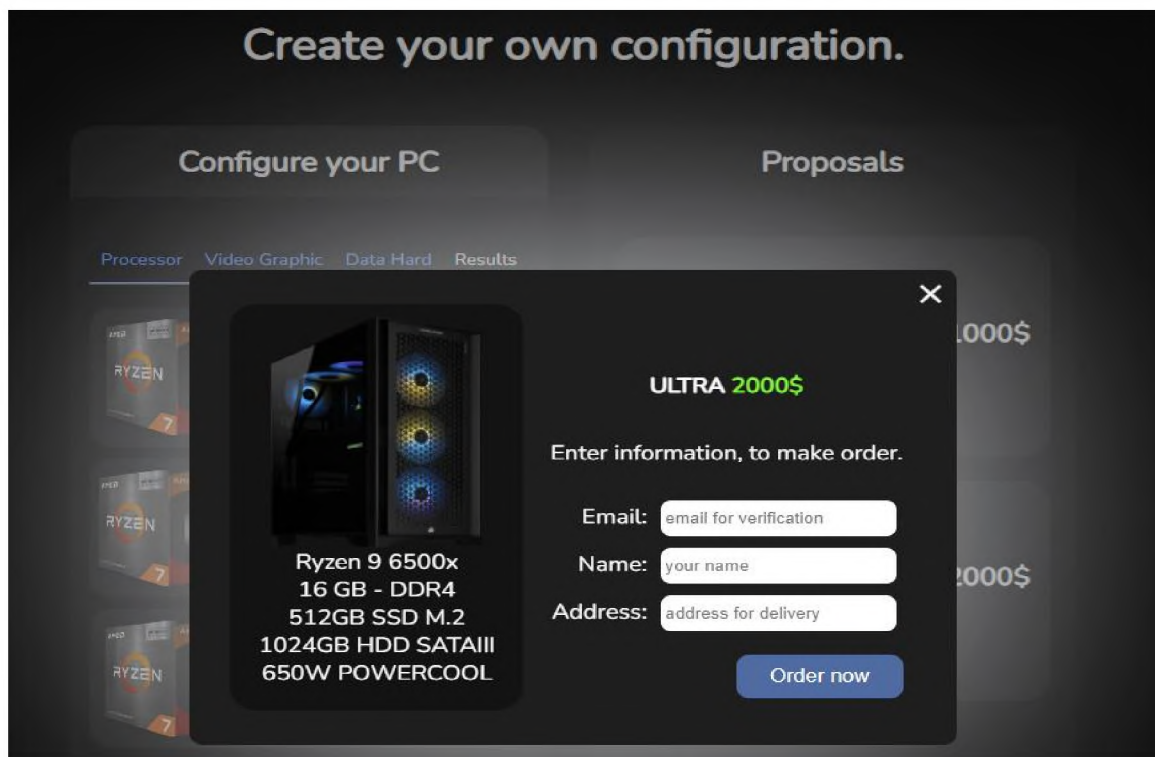


Рисунок 3.10 – Зображення працюючої форми для замовлення обраного ПК

Форма, яка наведена на рис. 3.10, обробляється спеціально створеною функцією на JavaScript, код якої наведено на рис. 3.11.

```
function buildPcBill(pc) {
  let item = pcList.get(parseInt(pc));
  return `<div class="show-item">
    <div class="left card">
      <div class="img" style="background-image:url('${item.image}')"></div>
      <div > ${item.descr}</div>
    </div>
    <div class="right">
      <div>
        <h4 class="head" style="margin-bottom:10px">${item.display} <b style="color:lime">${item.price}</b></h4>
        <span style="white-space:pre-line;">Enter information, to make order.</span>
      </div>
      <form class="form">
        Email: <input type="text" placeholder="email for verification">
        Name: <input type="text" placeholder="your name">
        Address: <input type="text" placeholder="address for delivery">
        <button class="btn" style='margin-top: 15px;'>Order now</button>
      </form>
    </div>
  </div>`;
}
```

Рисунок 3.11 – Лістинг функції для обробки даних форми

Для інтерактивної взаємодії користувача із сайтом були застосовані популярні прийоми проектування, зокрема трансформації. На рис. 3.12 показано, як збільшується зображення при наведенні покажчиком мишки.



Рисунок 3.12 – Збільшення (трансформація) зображення при наведенні

Для трансформації зображення використано спеціальні функції CSS і JavaScript, про що можна дізнатися з підказки над елементом у коді (рис. 3.13).

```

4      <div class="img" style="background-image: url('3s.png')"></div>
5      <div style="color: ■ gray">Price</div>
6
7      </div>
8      <div class="card pointer" data-show-pc="1">
9      <div class="title">S1Comp</div>
10     <div class="body">
11         <div class="img" style="background-image: url('3s.png')"></div>
12         <div style="color: ■ gray">Price</div>
13     </div>
14     <div class="footer">
15         S1Comp
16     </div>

```

A space-separated list of the classes of the element. Classes allows CSS and JavaScript to select and access specific elements via the [class selectors](#) or functions like the method `Document.getElementsByClassName()`.

[MDN Reference](#)

Рисунок 3.13 – Фрагмент опису коду одного із зображень, що трансформується

При виборі конфігурації ПК користувач обирає шляхом перемикання складових (процесор, відеокарта, тип накопичувача) потрібні моделі, а потім отримує два варіанти розрахунку із обраними пристроями (рис. 3.14).

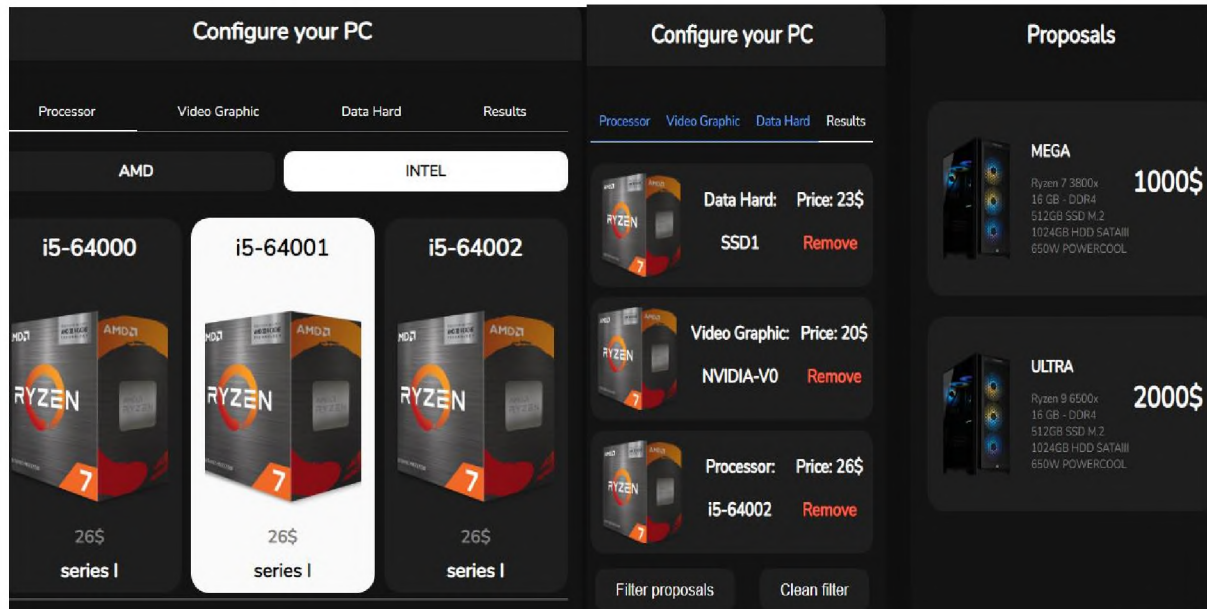


Рисунок 3.14 – Етапи реалізації вибору комплектуючих (зліва) та підрахунок вартості обраної конфігурації на сайті (справа)

Процедура підрахунку відбувається через функції JavaScript на основі визначеної ціни кожної деталі (рис. 3.15).

```

257     itemList.set(itemList.size, {
258         id: itemList.size,
259         display: "SSD" + index,
260         image: "s5.png",
261         descr: "series SD",
262         price: "23$",
263     });
264 }

```

Рисунок 3.15 – Фрагмент коду JS, в якому показано значення моделей і ціни кожної моделі накопичувача

Подібний (див. рис. 3.15) опис внесено по кожній номенклатурній деталі. Зрештою, відбувається формування вартості ПК в залежності від вибору користувача за допомогою спеціально прописаних функцій JS (рис. 3.16).


```

264     }
265     for (let index = 0; index < 21; ) {
266         setMapVal(categoryItems, "INTEL", itemList.get(0 + index));
267         setMapVal(categoryItems, "AMD", itemList.get(1 + index));
268         setMapVal(categoryItems, "INTEL-V", itemList.get(2 + index));
269         setMapVal(categoryItems, "NVIDIA-V", itemList.get(3 + index));
270         setMapVal(categoryItems, "AMD-V", itemList.get(4 + index));
271         setMapVal(categoryItems, "HDD", itemList.get(5 + index));
272         setMapVal(categoryItems, "SSD", itemList.get(6 + index));
273         index += 7;
274     }
275     document.addEventListener("DOMContentLoaded", (event) => {
276         pickerContent.forEach((el) => {
277             if (el.dataset.menuContent == pickerMenu[0].dataset.toggle) {
278                 switchInformation(pickerMenu[0].dataset.toggle);
279                 let sub = el.querySelector(".submenu li");
280                 el.querySelector(".items").innerHTML = buildCategory(
281                     sub.dataset.menu,
282                     sub.dataset.category
283                 ).join("");
284                 return;
285             }
286         });

```

Рисунок 3.16 – Фрагмент програми JS для розрахунку вартості ПК на основі комплектуючих, обраних користувачем

Як бачимо (див. рис. 3.16), враховані всі наявні пропозиції, верифікація здійснюється за допомогою операторів циклу For – Return та умовного переходу If – Let.

Для підтвердження замовлення та продовження оформлення покупки слугує форма, яка була описана вище (див. рис. 3.10). Отже, сайт є зручним, зрозумілим, функціональним.

3.3 Економічне оцінювання варіантів розробки вебсайту

У ході виконання кваліфікаційної роботи розроблено промо-сайт для ІТ-компанії, яка пропонує ознайомлення з комплектуючими, онлайн конфігування ПК, розрахунок вартості й оформлення замовлення. Здійснимо економічну оцінку вартості такої розробки для замовника, а також усереднений підрахунок затрат розробника.

Економічний аналіз – взаємопов’язані й взаємозумовлені методи вивчення і наукового дослідження певних явищ, процесів, дій, результатів. У економіці застосовується з метою виявлення закономірностей і тенденцій розвитку економічних процесів, встановлення та оцінки основних факторів, що позитивно чи негативно впливають на показники ефективності.

З боку розробника метою розробки будь якого ПЗ є отримання вигоди шляхом продажу продукту, тому необхідно провести економічний аналіз, який допоможе відповісти на такі питання, як ціна товару, переваги перед конкурентами і т. ін.

Для визначення вартості розробки вебсайту спочатку проведемо розрахунок трудомісткості, які включають в себе два окремих завдання:

1. Робота над UI-дизайном (дизайном зовнішнього виду) сайту.
2. Реалізація сайту за допомогою середовищ програмування.

Для розрахунку вартості реалізації подібного сайту необхідно врахувати такі показники як: вартість розробки дизайну, вартість реалізації сайту за допомогою інтегрованих середовищ розробки, вартість хостингу [38].

Найбільш важливим пунктом при розробці продукту є розрахунок його собівартості й визначення кінцевої ціни, оскільки саме вона має основну вагу при визначенні переваг над конкурентами та доцільності розробки системи.

При розрахунку затрат на проєктування враховуються такі показники як: оплата праці, комунікаційні витрати, інші витрати. Для розрахунків затрат на оплату праці візьмемо за основу, що проєктування і розробка виконується одним розробником. Він працює по стандартному графіку: восьмигодинний робочий день, п’ять днів на тиждень. Оплата праці – погодинна. Вартість години роботи може бути різною в залежності від наступних факторів:

- область застосування системи;
- обсяг застосовуваних інструментів та методів;
- обсяг первинної інформації по супроводу (документація);
- термін виконання;
- складність завдання.

У нашому випадку оплату праці приймаємо в середньому 25000 гривень за місяць. На розробку системи в нашому проєкті виділено 2 тижні (10 робочих днів) для професійного розробника.

Комунікаційні витрати представлені у вигляді плати за інтернет. Використовуємо безлімітний інтернет вартістю 300 грн на місяць. З розрахунку, що термін розробки сайту складає 14 днів, маємо витрати обсягом 140 грн.

Додатково в проєкт закладається очікувана норма запасу, яка рівна 20 %

За розробку сайту теж необхідно заплатити дизайнерам. Середня вартість розробки дизайну сайту, згідно тематичним сайтам в інтернеті, становить приблизно 10000 грн. Раніше було встановлено, що розробник буде працювати по 8 годинному графіку з оплатою 340 грн/годину. Формула розрахунку зарплати може бути записана так:

$$З_{\text{п}} = (K_{\text{днів}} \times V_{\text{роб}}) \times T_{\text{г}}, \quad (3.1)$$

де $K_{\text{днів}}$ – кількість робочих днів;

$V_{\text{роб}}$ – тривалість робочого дня в годинах;

$T_{\text{г}}$ – погодинна ставка розробника.

Підрахунок оплати праці розробника становить $З_{\text{п}} = (10 \times 8) \times 340 = 27200$ грн.

Загальний підрахунок витрат представлено в табл. 3.1.

Таблиця 3.1 – Розрахунок прямих витрат на розроблення вебсайту

№ з/п	Види витрат на розробку і впровадження технології	Витрати, од. часу	Грошовий еквівалент робіт, грн/од. часу	Сумарна вартість, грн
1	Розроблення макету вебсайту в програмі Figma (ПЗ безкоштовно)	24 год.	300	7200
2	Розроблення сайту (програміст)	80 год.	340	27200
3	Оплата домену (хостинг) від компанії TheHost, пакет «Діловий»	1 рік	799	799
4	Вартість домену (Міжнародний інформаційний домен Info) на рік	1 рік	249	249
5	Вартість послуг інтернет за період роботи (виготовлення сайту)	14 днів	10	140
6	Сумарна вартість індивідуального замовлення вебсайту	=(п.1+п.2+ п.3+ п.4+ п.5)		35588

Дані, які використані для підрахунку витрат на розроблення вебсайту, було взято з відкритих джерел. Зокрема, актуальні дані про хостинг взято із вебсайту компанії Thehost [39]. Відомості про середні розцінки роботи вебдизайнера, програміста публікує work.ua [40].

Таким чином, загальна вартість виготовлення вебсайту одноосібним програмістом або графічним дизайнером (макет сайту) і вебпрограмістом складе близько 35588 грн станом на початок 2024 р.

Об'єктивно, розроблення аналогічного вебсайту в професійній студії може бути дешевше, швидше. Це досягається за рахунок напрацьованого портфолію, розмежування видів робіт всередині команди виконавця. Якщо переглянути портфолію кожної студії, то можна відзначити, що на багатьох сайтах є повторювана структура, прийоми анімації та інше. Тому підрахунки, наведені в цьому розділі мають оціночний характер. Остаточне рішення приймає кожна компанія залежно від цілей, технічних та інших вимог.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було досліджено роль вебсайтів для бізнесу та систематизовано інформацію про існуючі технології виготовлення вебсайтів, сучасні тенденції дизайну, а також успішно створений промо-сайт для одного з напрямків IT-бізнесу, а саме сайт зі збирання ПК на основі наданих комплектуючих. За результатами роботи можна сформулювати такі висновки.

1. Базовими технологіями для розроблення вебсайтів різного спрямування були і залишаються HTML, який є основною мовою для створення структури вебсторінок та розміщення контенту; CSS, що можуть бути використані для стилізації вебсторінок та надання їм стилізованого вигляду; JavaScript – може бути використаний для динамічного створення вебсторінок, включаючи інтерактивність та анімацію.

2. Ефективність обраних методів залежить від їх правильного використання, відповідно до конкретних потреб та вимог проєкту. Важливо враховувати технічні можливості та обмеження, які можуть виникнути при розробці вебсайту. Для роботи з кодами вебсайту використано інтегроване середовище програмування Visual Studio Code, яке є лідером серед аналогічних програм для оптимального верстання.

3. Серед сучасних стилів дизайну для промо-сайту зручним та функціональним є стиль метро, який реалізовано в проєкті. Для створення дизайну та макету вебсайту використане гафічне середовище Figma.

4. Для коректного відображення на пристроях різного розміру (роздільної здатності моніторів) використано адаптивний дизайн за допомогою застосування flex-елементів та @media властивостей.

Розроблений сайт має стильний дизайн, оригінальну та інтуїтивно зрозумілу навігацію, а також містить корисну інформацію про компоненти для збирання ПК. Реалізований додаток відповідає цілям впровадження промо-

сайтів для бізнесу, а також конкретно визначеним завданням, поставленим перед реалізацією проєкту.

Економічне оцінювання вартості розроблення промо-сайту проведене за даними середньої вартості робіт на ринку України. За розрахунками вартість розроблення сайту під ключ з усіма виплатами й хостингом складатиме 30, 7 тис. грн.

Практична значимість такої розробки полягає як в її готовності до використання, так і в можливості удосконалення проєкту. Наприклад, можна додати актуальну інформацію про компоненти збірки ПК, щоб користувачі могли переконатися, що вони використовують найновіші та найкращі компоненти. Для цього можна використовувати різні API, які надають інформацію про компоненти, такі як PCPartPicker, Newegg або Amazon.

З часом можна покращити дизайн створеного сайту для кращого користувацького досвіду. Наприклад, додати на сайт актуальні контактні посилання, щоб користувачі могли легко зв'язатися з розробниками (представниками бізнесу), якщо у них виникнуть питання.

Окрім того, можна в подальшому автоматизувати приймання, оплату замовлень і доставку онлайн. Для цього можемо використовувати різні платіжні системи, такі як PayPal або Stripe, щоб користувачі могли швидко і легко здійснювати покупки на такому сайті. Можна також використовувати такі інструменти, як Zapier або Integromat, щоб автоматично отримувати сповіщення про замовлення та надсилати їх нашій команді для обробки.

Розроблений вебсайт може бути корисним як приклад для використання в навчальному процесі, а також як елемент особистого портфоліо розробника.