

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
Кафедра _____ Медіасистем та технологій _____
Рівень вищої освіти _____ перший (бакалаврський) _____
Спеціальність _____ 186 Видавництво та поліграфія _____
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Видавничо-поліграфічна справа _____
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри МСТ _____
(підпис)
« 22 » травня 2023 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студентові _____ *Крюковій Марині Михайлівні* _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ *Проектування та розробка web-сайту* _____
_____ *«Уроки створення сайту засобами HTML5 та CSS3»* _____

Затверджена наказом по університету від _____ 22 травня 2023 р. № 506 Ст _____


2. Термін подання студентом роботи до екзаменаційної комісії _____ 20 червня 2023 р. _____

3. Вихідні дані до роботи
Вид і призначення web-видання – інформаційний навчальний веб-сайт; мови розробки web-видання – HTML5, CSS3, JS; середовище розповсюдження – мережа Інтернет.

4. Перелік питань, що потрібно опрацювати в роботі
Вступ; Аналіз завдання на кваліфікаційну роботу; Аналітичний огляд досягнень у виробництві та застосуванні Web-видань; Вибір інструментальних засобів розробки; Проектування інформаційної структури та навігації; Розробка модульної сітки; Розробка графічного дизайну; Наповнення контентом сторінок видання; Тестування та публікація; Економічна частина; Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п. 5 включається до завдання за рішенням випускової кафедри)
Титульний слайд; Мета, задачі, актуальність роботи; Вибір інструментальних засобів розробки; Проектування інформаційної структури та навігації (2); Розробка модульної сітки; Розробка графічного дизайну (2); Наповнення контентом сторінок сайту; Тестування на кросбраузерність; Тестування на адаптивність; Публікація; Економічна частина; Висновки.

6. Консультанти розділів роботи

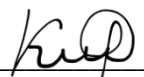
Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	проф. Єгорова І.М.		16.06.2023
Економічна частина	ас. Помогалова Н.В.		07.06.2023

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз завдання на кваліфікаційну роботу, визначення цілей і задач проектування	22.05	виконано
2	Аналітичний огляд у виробництві та застосуванні Web-видань	23.05	виконано
3	Вибір інструментальних засобів розробки	23.05	виконано
4	Проектування інформаційної структури та навігації	25.05	виконано
5	Розробка модульної сітки	27.05	виконано
6	Розробка графічного дизайну	28.05	виконано
7	Наповнення контентом сторінок видання	01.06	виконано
8	Тестування і публікація	04.06	виконано
9	Економічна частина	05.06	виконано
10	Оформлення пояснювальної записки	07.06	виконано
11	Оформлення графічної частини	10.06	виконано

Дата видачі завдання

Студент


(підпис)

Крюкова М.М.

Керівник роботи


(підпис)

проф. Єгорова І.М.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи містить: 48 с., 4 табл., 36 рис., 3 дод., 23 джерел.

ВЕБ-САЙТ, ВЕБ-РОЗРОБКА, УРОКИ, HTML5, CSS3, JAVASCRIPT, VISUAL STUDIO CODE, CODEPEN, НАВІГАЦІЯ, АДАПТИВНІСТЬ.

Метою кваліфікаційної роботи бакалавра є розробка веб-сайту «Уроки створення сайту засобами HTML5 та CSS3».

В ході роботи були виконані наступні дії: проаналізоване технічне завдання; проведений аналітичний огляд у виробництві та застосуванні веб-видань; обрані інструментальні засоби розробки веб-сайту; спроектовані інформаційна структура та навігація; розроблена модульна сітка; створений графічний дизайн; наповнені контентом сторінки веб-сайту; виконані тестування на кросбраузерність та на адаптивність; проведені розрахунки витрат на розробку веб-сайту.

У зв'язку із виконаними діями був розроблений адаптивний веб-сайт, який містить навчальну спрямованість та допоможе сформувати базові знання початківцям у веб-розробці.

ABSTRACT

The explanatory note of the qualification work contains: 48 p., 4 tabl., 36 pic., 3 app., 23 sources.

WEBSITE, WEB DEVELOPMENT, LESSONS, HTML5, CSS3, JAVASCRIPT, VISUAL STUDIO CODE, CODEPEN, NAVIGATION, ADAPTABILITY.

The purpose of the bachelor's qualification work is the development of the website "Lessons on Creating Website Using HTML5 and CSS3".

During the work, the following actions were performed: the technical task was analyzed; carried out an analytical review in the production and application of web publications; selected tools for website development; designed information structure and navigation; developed modular grid; created graphic design; content-filled website pages; cross-browser and adaptability tests were performed; calculations of website development costs were carried out.

In connection with the performed actions, an adaptive website was developed, which contains an educational focus and will help to form the basic knowledge for beginners in web development.

ЗМІСТ

	С.
ВСТУП	7
1 АНАЛІЗ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ.....	9
2 АНАЛІТИЧНИЙ ОГЛЯД ДОСЯГНЕНЬ У ВИРОБНИЦТВІ ТА ЗАСТОСУВАННІ WEB-ВИДАНЬ.....	11
2.1 Сучасні технології розробки веб-видань	11
2.2 Інструментальні засоби розробки веб-видань.....	12
2.3 Тестування веб-видань	14
3 ВИБІР ІНТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ.....	16
4 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СТРУКТУРИ ТА НАВІГАЦІЇ.....	19
5 РОЗРОБКА МОДУЛЬНОЇ СІТКИ.....	23
6 РОЗРОБКА ГРАФІЧНОГО ДИЗАЙНУ	25
7 НАПОВНЕННЯ КОНТЕНТОМ СТОРІНОК ВИДАННЯ.....	30
8 ТЕСТУВАННЯ ТА ПУБЛІКАЦІЯ	33
9 ЕКОНОМІЧНА ЧАСТИНА	40
ВИСНОВКИ.....	46
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	47
ДОДАТОК А Фрагмент коду html-документу.....	49
ДОДАТОК Б Фрагмент коду css-документу	51

ВСТУП

В наш час популярною сферою для роботи, до якої хочуть залучитися дедалі більше людей, є веб-розробка. У зв'язку з цим зростає попит на отримання знань в цій галузі. Звичайно, зараз людина в більшості випадків для знаходження інформації віддає перевагу веб-сайтам, в порівнянні з іншими носіями інформації, адже доступ до веб-видань є, найчастіше, вільним, швидким та легким. При цьому веб-сайти із навчальною спрямованістю користуються великою популярністю, особливо, коли на них розташована інформація, яка пов'язана з ІТ-сферою.

Актуальність обраної теми може бути підтверджена тим, що існує значний попит серед користувачів на отримання ключових знань зі створення веб-сайтів, що зібрані в одному виданні. Вивчаючи аналоги, було з'ясовано, що веб-видань саме про створення сайтів вкрай мало. Проаналізувавши аналоги сайтів із відомостями про HTML та CSS, було з'ясовано, що вони містять інформацію про певні елементи, а демонстрація застосування цих елементів обмежується лише окремими прикладами. При цьому користувачам-початківцям, більш важливим є з'ясування як ці елементи виглядають в загальному коді веб-сайту, тобто в природному середовищі, а не в штучному форматі. Також було звернено увагу на дизайн аналогів: в більшості випадків він має несучасний вигляд, що, звичайно, може відштовхнути користувачів.

Метою кваліфікаційної роботи є створення веб-сайту на тему «Уроки створення сайту засобами HTML5 та CSS3», орієнтуючись на вміст якого, початківці у веб-розробці отримають знання стосовно створення свого першого сайту, використовуючи базові коди-будівельники сайтів – HTML5 та CSS3.

Веб-видання буде містити 10 сторінок, із Головною включно. З Головної сторінки можна буде перейти на дев'ять пропонованих уроків, які знаходяться на окремих сторінках. Більшість уроків буде присвячена базовим елементам

HTML – його структурним елементам, тексту, спискам, посиланням, зображенням, таблицям, формам. Також будуть уроки стосовно стильового оформлення сторінки із застосуванням CSS-коду.

В якості інструментальних засобів для написання коду для проєктованого сайту обране програмне забезпечення Visual Studio Code, в якому планується використання мов HTML, CSS та JavaScript. Для проєктування графічного дизайну сайту обрана програма Figma, для кольорової корекції зображень сайту – Adobe Photoshop.

По завершенню проєктування планується отримати результат у вигляді адаптивного веб-видання, яке буде містити якісну та зручно подану інформацію.

1 АНАЛІЗ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Метою розробки є створення веб-сайту на тему «Уроки створення сайту засобами HTML5 та CSS3», орієнтуючись на вміст якого, початківці у веб-розробці отримають знання стосовно створення свого першого сайту, використовуючи базові коди-будівельники сайтів – HTML5 та CSS3.

Для досягнення поставленої мети необхідно виконати ряд завдань:

- вибір інструментальних засобів розробки;
- проектування інформаційної структури та навігації;
- розробка модульної сітки;
- розробка графічного дизайну;
- наповнення контентом сторінок веб-видання;
- тестування.

До потенційної цільової аудиторії проєктованого сайту належать користувачі Інтернету будь-якої статі, основним віком від 16 до 26 років. Відповідно, це учні старших класів та студенти, яких цікавлять веб-технології та які прагнуть приєднатися до галузі веб-розробки та сформувати певну базу знань щодо цієї теми, щоб в подальшому їм було легше працевлаштуватися в цій галузі.

Проектований веб-сайт має бути адаптований до різних пристроїв: комп'ютерів, планшетів, смартфонів, адже в наш час користувачі переглядають сайти на будь-яких екранах та без застосованої адаптивності до сайту, можна втратити певну кількість користувачів, яким буде не зручно користуватися сайтом. Таким чином, теоретичну інформацію на проєктованому сайті користувачі можуть переглядати на різних пристроях, але для освоєння практичної інформації із демонстрацією коду, краще використовувати комп'ютер, адже пристроєм для створення коду є саме він. Проектований веб-сайт також не повинен мати велику кількість зображень. Це може сприяти більш довгому завантаженню сторінок, що не вітається на інформаційних сайтах, де найбільш важливим є текстовий контент.

Потенційним користувачам на проєктованому сайті буде надана інформація щодо основ HTML5 та CSS3, яка розподілена на певну кількість уроків. В кожному уроці можна буде ознайомитися з блоками теорії та практики. В блоці із практикою, вивчаючи кожний урок, можна буде слідкувати за поетапним створенням нескладного односторінкового веб-сайту. Буде демонструватися вбудований в сторінку CodePen з інтерактивним кодом. CodePen – це соціальне середовище для розробників та дизайнерів фронтенду, де можна переглядати та взаємодіяти з кодом інших розробників, і, звичайно, створювати свої коди. CodePen також надає можливість вбудовувати код на інші веб-сайти. Вбудований Pen є потужним інструментом для відображення демонстраційного коду: можна не тільки бачити реальний результат, але й побачити весь код, який використовувався для його створення. Крім того, користувачі можуть редагувати вбудований код, що додає певну інтерактивність [1].

При розробці сайту планується використання стандартних технологій, а саме мов HTML5, CSS3 та JavaScript, яких достатньо для створення веб-сайту із необхідним функціоналом. Як було зазначено вище, на сторінку, в HTML-код буде вбудовуватися інтерактивний код CodePen, який добре поєднується з обраними технологіями.

2 АНАЛІТИЧНИЙ ОГЛЯД ДОСЯГНЕНЬ У ВИРОБНИЦТВІ ТА ЗАСТОСУВАННІ WEB-ВИДАНЬ

2.1 Сучасні технології розробки веб-видань

Сучасними базовими технологіями розробки веб-сайтів, які повинні знати усі веб-розробники, є HTML5, CSS3 та JavaScript.

HTML5 – це остання версія мови розмітки HTML, що пропонує кілька нових і корисних можливостей для сучасної веб-розробки. Серед них – семантичні елементи, які дозволяють більш точно визначити структуру сторінки, вбудована підтримка мультимедіа, офлайн-зберігання, покращена підтримка форм та елементів вводу. HTML5 дозволяє створювати більш інтерактивні, доступні та естетично привабливі веб-сторінки та додатки [2].

CSS3 є останньою версією мови CSS, яка пропонує ряд нових можливостей для стилізації веб-сторінок. Вона включає гнучкіший вибір селекторів, можливість застосовувати анімацію та перехідні ефекти, покращені можливості роботи з тіннями та градієнтами, можливість закруглювати кути елементів, підтримку шрифтів іконок, гнучкішу роботу з фоновими зображеннями, можливість використовувати властивості для медіа-запитів та респонсивного дизайну. CSS3 надає розробникам більші можливості для стилізації та візуального оформлення веб-сторінок, дозволяючи створювати ефектні та привабливі дизайни.

Медіа-запити, можливо, є найбільшим чинником змін у дизайні веб-сайтів протягом багатьох років. Широке поширення CSS дозволило залишити у минулому жорсткі обмеження веб-сайтів на основі таблиць, а медіа-запити зробили ще один крок вперед, дозволяючи стилізувати сторінки таким чином, щоб вони відповідали специфікаціям кожного пристрою, на якому відображаються сайти. Завдяки ним можна задавати різні стилі для екранів з різною шириною, встановлювати інші розміри шрифтів або показувати чи

ховати деякі елементи залежно від типу пристрою. Це дозволяє забезпечити більш адаптивний та зручний дизайн веб-сторінок для різних користувачів та пристроїв. Медіа-запити є потужним інструментом для розробників, які дозволяють створювати респонсивні та мобільні версії веб-сайтів [3].

Також слід виділити технології CSS3 – Flexbox та Grid Layout, які дозволяють створювати різноманітні та гнучкі макети веб-сторінок.

Технологія Flexbox стала в своєму роді революційною при верстці сторінок сайтів. Вона дозволила позбутися від великої кількості проблем, пов'язаних з вирівнюванням елементів, їх масштабуванням при зміні розмірів екранів, таким чином, забезпечив реалізацію адаптивності розробки. Основною концепцією даної технології стало автоматичне обчислення інтервалів між елементами, що спростило заповнення контейнера-батька.

Технологія CSS Grid – найновіша з нині застосовуваних у верстці. В ній використані принципи табличної верстки, реалізовані засобами CSS. Дана технологія дозволяє розділяти макет типу таблиці, не торкаючись при цьому HTML розмітки. На відміну від Flexbox, CSS Grid працює одразу з двома осями. Ця технологія дозволила застосовувати нові способи розміщення блоків, надаючи дизайнерам ще більшу свободу дій [4].

Ще одною сучасною технологією розробки веб-сайтів є, як було зазначено вище, JavaScript. JavaScript – це мова програмування, яка дозволяє наповнити HTML анімацією, інтерактивністю та динамічними візуальними ефектами. Однією з головних переваг JavaScript є її миттєвість. Це дозволяє веб-сторінкам миттєво реагувати на такі дії, як натискання посилання, заповнення форми або просто переміщення миші по екрану [5, 6].

2.2 Інструментальні засоби розробки веб-видань

Для написання коду HTML, CSS, JavaScript найпоширенішою програмою зараз виступає Visual Studio Code. Visual Studio Code – це потужний редактор коду з відкритим вихідним кодом, який працює у Windows, Linux та macOS.

Цей програмний продукт підтримує велику кількість мов, зокрема Python, Go, Java, JavaScript, Node.js, Ruby, HTML, CSS, C#. Функції включають підтримку налагодження, підсвічування синтаксису, фрагменти коду, рефакторинг коду та вбудований Git. Користувачі можуть змінити тему, сполучення клавіш, налаштування та встановити розширення, що додають функціональність [7, 8]. Ще одною корисною функціональністю VS Code є інтелектуальне завершення коду – IntelliSense. Функції VS Code IntelliSense працюють на основі мовної служби. Мовна служба забезпечує інтелектуальне завершення коду на основі семантики мови та аналізу вихідного коду. IntelliSense здійснюється у вигляді спливаючих вікон автозавершення під час введення тексту, запиту параметрів функцій і підказок запиту, пов'язаних із синтаксичними помилками [9]. Також при розробленні веб-сайту задіюються і графічні редактори, які допомагають при макетуванні сторінок сайту та при розробці графічного дизайну.

Для створення макетів веб-сторінок та інтерфейсів зараз користується великою популярністю програма Figma. Figma – це спільний веб-додаток для розробки інтерфейсу з додатковими офлайн-функціями, доступними в настільних програмах для macOS і Windows. Усі файли Figma зберігаються не на комп'ютері, а в хмарі, що дозволяє легко синхронізувати та отримувати до них доступ з різних пристроїв. Набір функцій Figma зосереджений на дизайні інтерфейсу користувача та взаємодії з користувачем, з акцентом на співпрацю в реальному часі, використовуючи різноманітні інструменти для створення векторної графіки та прототипів. Figma також має потужні можливості роботи з текстом, такі як вибір шрифтів, налаштування розміру та керування оформленням тексту. Дана програма дозволяє створювати стилі для кольорів, шрифтів та елементів дизайну, що дозволяє легко застосовувати їх до різних елементів у макеті. Figma надає інструменти для створення сіток, що допомагає створювати збалансовані та консистентні макети з правильним розміщенням елементів. Figma містить панель Inspect, завдяки якій можна подивитися та скопіювати CSS-код будь-якого елементу, що спрощує їх

вставку на веб-сторінку сайту. Також Figma містить велику кількість плагінів, які дозволяють автоматизувати певні завдання та покращити робочий процес [10].

При графічному оформленні веб-сайтів також широко застосовується програма Adobe Photoshop. Це растровий редактор, який у першу чергу спеціалізується на редагуванні та покращенні фотографій. Photoshop надає потужний інструментарій для кольорокорекції зображень, їх ретуші та кадрування. Також Photoshop дозволяє об'єднувати різні фотографії та графічні елементи для створення композицій. Завдяки цій програмі можна створювати колажі, вирізати об'єкти з фотографій, налаштовувати прозорість шарів та застосовувати різноманітні ефекти [11].

Для створення векторної графіки веб-дизайнери також застосовують програмний продукт Adobe Illustrator. Illustrator забезпечує велику кількість інструментів для маніпуляції формами, лініями, кольорами та текстом. Він має розширені можливості управління кривими Без'є, що дозволяє зручно налаштовувати форми об'єктів. Також в новій версії цієї програми з'явилася функція конвертації 2D-графіки в 3D із додаванням текстур та матеріалів [12]. Це розширює можливості веб-дизайнерів, адже 3D-графіка на веб-сайтах зараз є дуже поширеною.

2.3 Тестування веб-видань

Тестування веб-сайтів є важливим етапом в процесі їх розробки та підтримки. Його основною метою є перевірка функціональності, ефективності, безпеки та коректності роботи перед використанням веб-сайту користувачами. Серед багатьох видів тестувань можна виділити два суттєвих типи – на адаптивність та кросбраузерність, адже з поширенням різних браузерів та пристроїв, важливо переконатися, що веб-сайт коректно відображається на різноманітних екранах.

Тестування на адаптивність дозволяє перевірити, наскільки добре веб-сайт працює на різних типах пристроїв, включаючи комп'ютери, планшети та

смартфони. При тестуванні потрібно переконатися, що веб-сайт послідовно відображає та вирівнює вміст, відтворює текст розбірливо в усіх масштабах, зберігає вміст (текст та зображення) у своїх контейнерах, відображає та змінює розмір зображень за потреби, дозволяє користувачам прокручувати вертикально (або горизонтально, як у випадку адаптивних таблиць даних), дозволяє користувачам переміщатися за допомогою посилань і меню на всіх пристроях [13]. При тестуванні на адаптивність зараз широко застосовуються інструменти розробника, а саме DevTools, які вбудовані в усі сучасні браузері, зокрема, Firefox та Chrome DevTools. В інструментах розробника натискається сполучення клавіш Ctrl+Shift+M та відкривається панель, на якій можна обрати один з популярних мобільних пристроїв, що дозволить миттєво переглянути, як сайт відображається на них. Також є можливість змінювати роздільну здатність вручну, щоб оцінити продуктивність веб-сайту на власних роздільних здатностях [14].

Підтримка крос-браузерності, також відома як крос-браузерна сумісність, стосується здатності веб-сайту, конструкцій HTML підтримувати кілька веб-браузерів. Це дозволяє веб-сайту правильно та однаково відтворюватися в усіх браузерах. Сумісність браузера перевіряється для таких областей, як: макет сторінки, навігація, колір та графіка, мультимедіа, функціональність. Як правило, для перевірки застосовують декілька популярних браузерів: Google Chrome, Mozilla Firefox, Opera [15].

3 ВИБІР ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ

При проектуванні веб-видання необхідно підібрати доцільні та найбільш підходящі інструменти, щоб отримати бажаний результат.

Для вибору інструментального засобу для верстання веб-сайту проведено порівняння декількох програмних забезпечень для набору коду. Було обрано три найбільш популярних редакторів коду: Visual Studio Code, Sublime Text, Atom. В якості критеріїв для порівняння були обрані суттєві показники, які є важливими при проектуванні веб-сайту: «Уроки створення сайту засобами HTML5 та CSS3». Результати порівняння представлені в таблиці 3.1.

Таблиця 3.1 – Порівняння інструментальних засобів редагування коду

	Visual Studio Code	Sublime Text	Atom
Безкоштовність	+	-	+
Швидкість	8	10	6
Простота використання	9	10	8
Функція автозаповнення коду	10	6	7

Виходячи з даних в таблиці 3.1, можна побачити, що більшу кількість балів набрав редактор коду Visual Studio Code. Ця програма є безкоштовною, як і Atom, але по трьом іншим показникам вона є кращою за програму Atom. У той же час редактор коду Sublime Text має високу швидкість через те, що має дуже простий інтерфейс, який не перевантажений купою функцій. Відповідно, він є найпростішим у використанні. Але Visual Studio Code при цьому має широкий функціонал можливості автозаповнення, а саме функцію IntelliSense, чого не має у двох інших програм. Це важливий критерій, адже за допомогою цієї функції, яка має можливості автозаповнення, автоматичного відображення підказок, властивостей, методів, типів даних можна значно скоротити час розробки сайту. Дані можливості дозволяють розробникам

швидше знаходити потрібні команди, уникати помилок та підвищувати продуктивність при розробці. Також програма Sublime Text є платною, що, навіть при вагомій конкуренції із Visual Studio Code, не дозволяє її обрати в даному проектуванні. Таким чином, був обраний інструментальний засіб Visual Studio Code для верстки сайту. Мови, які будуть застосовані в цьому програмному забезпеченні – HTML, CSS та JavaScript.

Також при розробці веб-сайту необхідна програма для макетування майбутніх сторінок проектованого сайту. Для порівняння були обрані дві популярні програми – Figma та Adobe XD. Результати порівняння занесені в таблицю 3.2.

Таблиця 3.2 – Порівняння інструментальних засобів для створення макетів сайту

	Figma	Adobe XD
Можливість працювати офлайн	-	+
Більш використана та доступна веб-бібліотека шрифтів	+	-
Інтеграція з Photoshop	-	+
Гнучкіша організація елементів дизайну	+	-
Зручніша генерація коду CSS	+	-

Як можна побачити із таблиці 3.2, Adobe XD випереджає Figma у технічних аспектах, а саме: можливості працювати офлайн та інтеграцією із програмою Photoshop, яка також буде задіяна при проектуванні веб-сайту. Однак Figma випереджає Adobe XD по більш важливим критеріям для макетування сторінок – можливостям дизайну. На відміну від Adobe XD, Figma має найбільш використовувану веб-дизайнерами дуже потужну бібліотеку із великою кількістю безкоштовних шрифтів – Google Fonts. Для організації елементів дизайну Figma використовує фрейми, в той час як Adobe XD – артборди. Фрейми схожі на артборди, але вони можуть використовуватися як вікна для перегляду, так і як під-елементи дизайну: кнопки, секції контенту, навігаційні елементи тощо. Таким чином, фрейми

можна вкладати один в одного, створюючи ієрархію розміщення елементів на макеті, що дозволяє логічно організувати дизайн та швидко маніпулювати окремими елементами. Також Figma має зручну панель шарів, на відміну від Adobe XD. Шари в Figma підтримують вкладеність та виділяються при наведенні на елементи в робочій області, що є дуже зручним. Для кожного елементу робочої області в Figma можна подивитися код CSS в панелі Inspect, який оновлюється в режимі реального часу. Це є значно зручнішим, ніж отримання коду в процесі генерації посилання на нього в Adobe XD, а потім відвідування його в браузері. Хоча Figma не має офлайн доступу, адже це хмарна програма, з іншого боку, файли даної програми не займають пам'ять комп'ютера, що є позитивним. Отже, орієнтуючись на більш гнучкіші можливості дизайну, була обрана програма Figma для макетування сторінок майбутнього сайту.

Також, враховуючи те, що на сайті будуть присутні зображення, необхідно обрати програму для їх кольорокорекції. Розуміючи, що в даному проектуванні кольорокорекція необхідна сама базова – налаштування насиченості, тону та контрастності, була обрана класична та найпопулярніша програма для редагування растрових зображень – Adobe Photoshop. Дана програма має усі необхідні інструменти для професійної кольорокорекції зображень.

4 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СТРУКТУРИ ТА НАВІГАЦІЇ

Інформація відіграє важливе значення у сучасному світі. Вона стає основою функціонування різних сфер діяльності, розвитку, економіки. Інформація займає лідируючі позиції, коли все переходить на цифрові технології [16]. Інтернет та його складові – веб-сайти є безцінним джерелом інформації, якою ділиться весь світ. Але ж сама наявність інформації на веб-сайті не робить його відразу безцінним. Необхідно, щоб інформація була певним чином структурована для зручного її споживання користувачами. Для цього вона розбивається на логічні блоки, будь то глави, розділи або сторінки.

Два основні типи інформаційної структури сайту – це лінійна (послідовна) та нелінійна (ієрархічна). При лінійній структурі інформація на веб-сайті має чітко визначений шлях та користувач переміщується з однієї сторінки на іншу в певній послідовності. Нелінійна структура передбачає організацію інформації на веб-сайті у вигляді древовидної структури, де кожна сторінка може мати гілки та підсторінки. Користувач може переміщатися по сайту, переходячи між різними розділами та підрозділами, не обмежуючись жорсткою послідовністю. Більшість сайтів використовує і ієрархічну, і послідовну структури, тобто, мають комбіновану структуру [17].

Інформаційна структура розроблюваного сайту є комбінованою. Проектований сайт містить уроки, кожний з яких розташований на окремій веб-сторінці. Блок із посиланнями на усі уроки знаходиться на Головній сторінці сайту. Головна сторінка сайту являє собою стартову точку та містить посилання на інші сторінки з уроками. Це вказує на наявність нелінійних елементів, тому що користувач може вибрати конкретний урок, який його цікавить, і перейти до нього безпосередньо, не переглядаючи попередніх або наступних уроків. Але при цьому краще все ж таки вивчати уроки послідовно. Для цього на кожній сторінці уроку є посилання на попередній і наступний урок, що створює певну послідовність і вказує на лінійні елементи. Це

дозволяє користувачам переглядати уроки послідовно, переходячи від одного уроку до іншого.

Кожний урок проєктованого веб-сайту містить певну структуру: спочатку йде блок з теорією, а потім – із практикою. В блоці із теорією знаходяться основні дані – заголовки та основний текст, відповідно до теми уроку, які можуть супроводжуватися графічними зображеннями та прикладами окремого коду. В блоці із практикою демонструється код сайту, який створюється, закріплюючи знання з теоретичного блоку. З кожним уроком, відповідно, код стає все більше. Також в цьому блоці описуються основні моменти створеного коду.

Для знаходження інформації користувачами на сайті проєктується навігація. Створення навігації – це дуже відповідальний крок, адже без зручно розташованої навігації та без швидкого доступу до неї користувач буде дезорієнтований, і, скоріш за все, ненадовго залишиться на сайті.

При проєктуванні сайту була розроблена навігаційна схема, яка продемонстрована на рисунку 4.1. Таким чином, сайт містить 10 сторінок, включно із Головною. На останніх дев'яти сторінках розташовуються уроки по HTML5 та CSS3.



Рисунок 4.1 – Навігаційна схема проєктованого веб-сайту

Навігація на проєктованому веб-сайті здійснюється за допомогою гіперпосилань. На кожній сторінці у верхній частині сайту розташована горизонтальна навігаційна панель, за допомогою посилань якої можна перейти на Головну сторінку та до списку уроків, які також знаходяться на Головній сторінці. Список уроків можна назвати основним навігаційним блоком даного

сайту, адже назва кожного уроку є гіперпосиланням на окрему сторінку сайту із певним уроком. Верхня навігаційна панель є липкою, тобто вона залишається видимою при прокручуванні сторінки вниз, що забезпечує постійний доступ до основних навігаційних елементів сайту. Ця функція реалізована за допомогою мови JavaScript. Підвал сайту також містить посилання на Головну сторінку та Уроки.

Після кожного уроку присутні кнопки, при натисканні на які можна перейти до попереднього та наступного уроку. Таким чином, користувач може перейти до необхідного уроку максимум за два кліки, використовуючи або посилання на попередній/наступний урок, або за допомогою посилання з верхньої навігаційної панелі до блоку усіх уроків. Також за допомогою JavaScript реалізована кнопка із стрілкою, при натисканні на яку буде здійснюватися перехід на вершину сторінки. В самій верхній частині сайту присутній індикатор прокрутки, створений за допомогою JavaScript. Цей елемент відображає, наскільки близько к кінцю сторінки знаходиться користувач, що спрощує перегляд довгих сторінок, які також присутні на проєктованому сайті. При перегляді сайту з телефону головне меню трансформується у гамбургер-меню, пункти якого можна відкрити натисканням кнопки.

Приклади навігаційних елементів проєктованого веб-сайту приведені на рисунках 4.2-4.4.



Рисунок 4.2 – Горизонтальна навігаційна панель

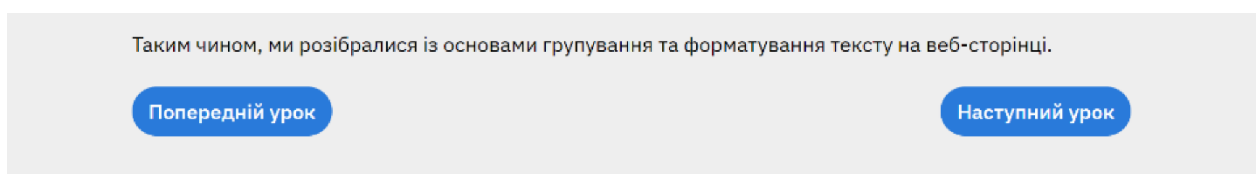


Рисунок 4.3 – Кнопки для переходу між сусідніми уроками

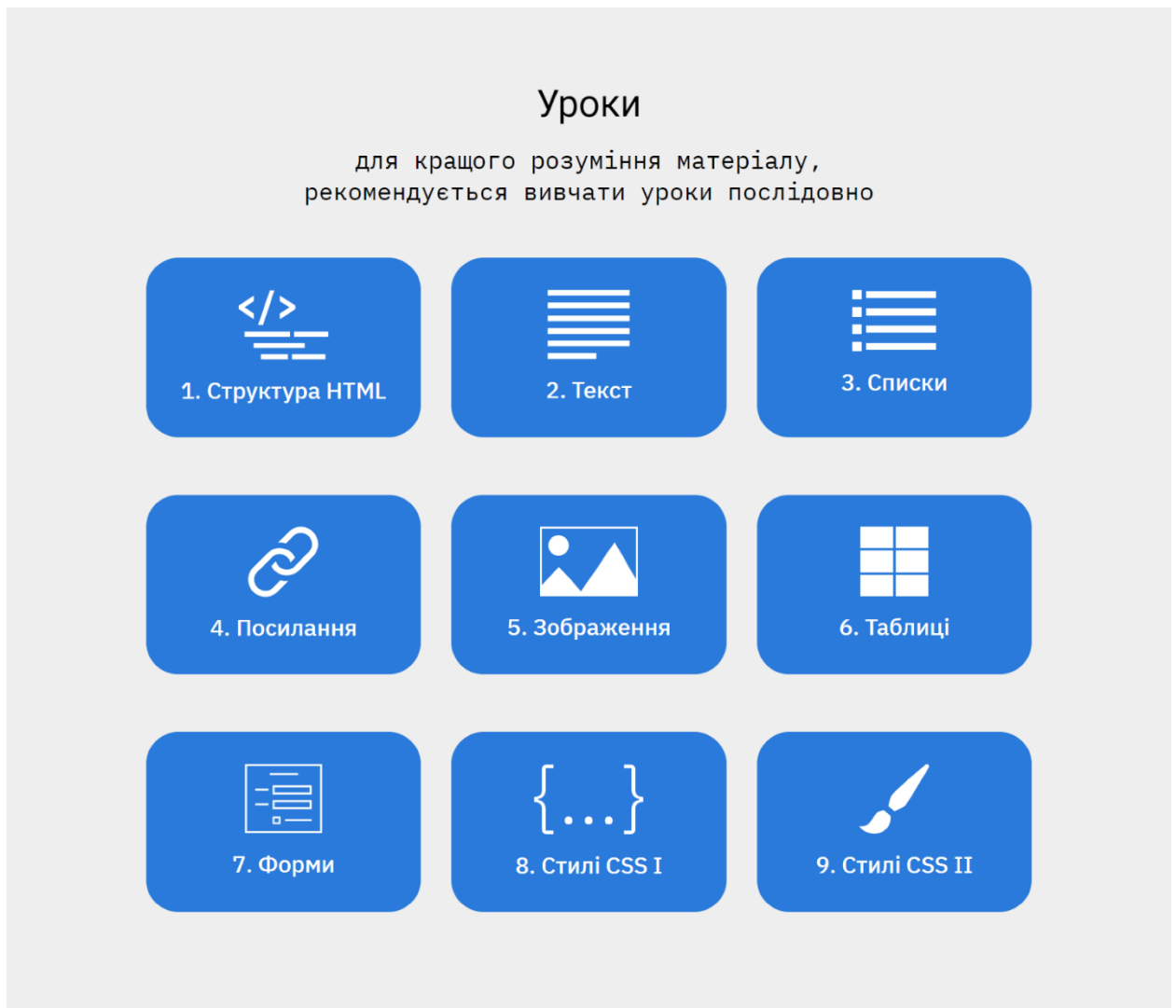


Рисунок 4.4 – Блок із посиланнями на усі уроки сайту

5 РОЗРОБКА МОДУЛЬНОЇ СІТКИ

Модульна сітка є набором невидимих направляючих, уздовж яких розташовуються елементи Web-сторінки. Це полегшує розміщення даних в документі, забезпечує візуальний зв'язок між окремими блоками і зберігає спадкоємність дизайну при переході від однієї сторінки до іншої. WEB-сторінка фактично розглядається як набір прямокутних блоків, які викладаються в певному порядку [18].

В веб-дизайні вертикальна сітка є більш важливою та легшою для визначення, ніж горизонтальна. Це пов'язано з природою веб-дизайну та концепцією, що висота сторінки значною мірою залежить від довжини вмісту. Таким чином, хоча горизонтальне вирівнювання бере участь у багатьох аспектах веб-дизайну, вертикальні лінії сітки, тобто стовпці, створюють основу для структури сітки. [19].

Найпоширенішою кількістю стовпців для сітки веб-сайту є 12. Завдяки цьому числу можна легко організувати вміст у будь-якому числі стовпців, кратних 12. Це означає, що використання 12-колонкової сітки дозволяє зручно створювати макети з 2, 3, 4 або 6 стовпцями.

В проєктованому сайті кількість колонок варіюється від 1 до 3 та для організації контенту була обрана 12-колонкова сітка. Сітка проєктованого сайту представлена на рисунку 5.1, а приклад її використання – на рисунку 5.2.

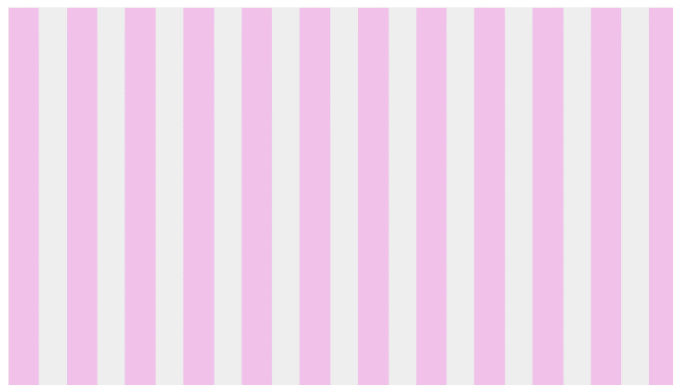


Рисунок 5.1 – Сітка проєктованого веб-сайту



Рисунок 5.2 – Приклад застосування сітки на проектованому веб-сайті

6 РОЗРОБКА ГРАФІЧНОГО ДИЗАЙНУ

Одним із творчих кроків при проектуванні веб-сайту є його оформлення, а саме створення графічного дизайну. При цьому слід приділити увагу вирівнюванню та конфігурації контенту веб-сторінки, колірній гамі, шрифту, логотипу, іконкам, декоративним елементам.

На проєктованому сайті використовується вирівнювання тексту по лівому краю та по центру. По центру вирівнюються заголовки та невеликі блоки тексту із декількох рядків, які знаходяться на деяких сторінках під заголовками. Основний текстовий контент на кожній сторінці, присвяченій певному уроку, має виключку по лівому краю. Вирівняний ліворуч текст дозволяє уникати непотрібних рухів очей, що покращує читабельність контенту. Кожна сторінка проєктованого сайту із уроками має однакову структуру: вона поділена на два блоки – інформаційну та практичну частину, внутрішні та зовнішні відступи в яких ідентичні, що призводить до стабільності та гармонічного сприйняття контенту проєктованого веб-сайту.

Колірне рішення сайту містить в собі декілька поєднань кольорів. На рисунку 6.1 представлені кольори, які зустрічаються на проєктованому сайті, в порядку зменшення займаної площі ними. Згідно з результатами досліджень, молода аудиторія, віком від 16 до 30 років, віддає перевагу кольорам, які створюють відчуття комфорту. Їй подобаються чисті, природні, живі та свіжі кольори [20].



Рисунок 6.1 – Колірна схема проєктованого веб-сайту

Навігаційні блоки, які містять в собі посилання, розроблені в світло-блакитному кольорі та відтінках синього для акцентування уваги. Ці кольори

мають певні асоціації з комп'ютерними технологіями, також вони створюють приємне та спокійне візуальне середовище для користувачів, що сприяє легкій орієнтації по сайту. Блакитний колір знижує м'язову напругу, освіжає, асоціюється з чистотою, зосереджує увагу. Синій колір сприяє розвитку розумових здібностей та поліпшує пам'ять. Теоретична частина уроків виконана в м'ятному фоновому кольорі. Даний колір підвищує м'язову працездатність на довгий час, надає відчуття бадьорості, чинить освіжаючий та одночасно заспокійливий вплив на організм [21]. Блоки із практикою містять нейтральний сірий фоновий колір для контрастування із теорією та для створення певного колірного балансу. Сірий є кольором інтелекту, він одночасно розслабляє та заспокоює [22]. Із теплих кольорів на сайті присутній пісочний, щоб розбавити холодні та нейтральні відтінки та в певних місцях акцентувати увагу, наприклад, для виділення підзаголовків в теоретичних блоках уроків.

Кожна сторінка проектованого веб-сайту починається із абстрактного фонового зображення, на тлі якого представлено зображення із закругленими кутами, яке пов'язане із тематикою уроку. Оформлення сторінок із уроками по HTML трохи відрізняється від уроків по CSS. В останніх уроках виділення оформлення певних деталей тексту та перших екранів сторінок є більш яскравими, в протилежність більш стриманому оформленню уроків по HTML, щоб підкреслити смислову спрямованість уроків.

Щодо шрифтового оформлення, на сайті використано три гарнітури, які представлені на рисунках 6.2-6.4. Для елементів меню та заголовків секцій сайту використана гротескна гарнітура Roboto. Для заголовків використано накреслення Medium, для елементів навігації – Regular. Для тексту, який знаходиться в деяких секціях під заголовком використана моноширинна гарнітура IBM Plex Mono, яка має певну асоціацію із набором коду. Для підзаголовків та основного тексту використана гротескна гарнітура IBM Plex Sans, із накресленнями Medium та Regular, відповідно. Обрані гарнітури виглядають сучасно, чітко та розбірливо.

Roboto. Regular & Medium

АБВГГДЕЕЖЗИІЙКЛМНОПРСТУФХЦЧШЩЬЮЯ
 абвггдеежзиіійклмнопрстуфхцчшщьюя
 0123456789!"#\$%&'()*+,-./:;<=>@[\\]^_`{|}~

АБВГГДЕЕЖЗИІЙКЛМНОПРСТУФХЦЧШЩЬЮЯ
 абвггдеежзиіійклмнопрстуфхцчшщьюя
 0123456789!"#\$%&'()*+,-./:;<=>@[\\]^_`{|}~

Рисунок 6.2 – Гарнітура Roboto

IBM Plex Mono. Regular

А Б В Г Г Д Е Є Ж З И І Й К Л М
 Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ю Я

а б в г г д е є ж з и і й к л м
 н о п р с т у ф х ц ч ш щ ь ю я

0 1 2 3 4 5 6 7 8 9 ! " # \$ % & '
 () * + , - . / : ; < = > ? @ [\]
 ^ _ ` { | } ~

Рисунок 6.3 – Гарнітура IBM Plex Mono

IBM Plex Sans. Regular & Medium

АБВГГДЕЕЖЗИІЙКЛМНОПРСТУФХЦЧШЩЬЮЯ
 абвггдеежзиіійклмнопрстуфхцчшщьюя
 0123456789!"#\$%&'()*+,-./:;<=>@[\\]^_`{|}~

АБВГГДЕЕЖЗИІЙКЛМНОПРСТУФХЦЧШЩЬЮЯ
 абвггдеежзиіійклмнопрстуфхцчшщьюя
 0123456789!"#\$%&'()*+,-./:;<=>@[\\]^_`{|}~

Рисунок 6.4 – Гарнітура IBM Plex Mono

Для проєктованого сайту був створений логотип, який приведений на рис. 6.5. Логотип є мінімалістичним, сучасним, символізучим код веб-розробки.

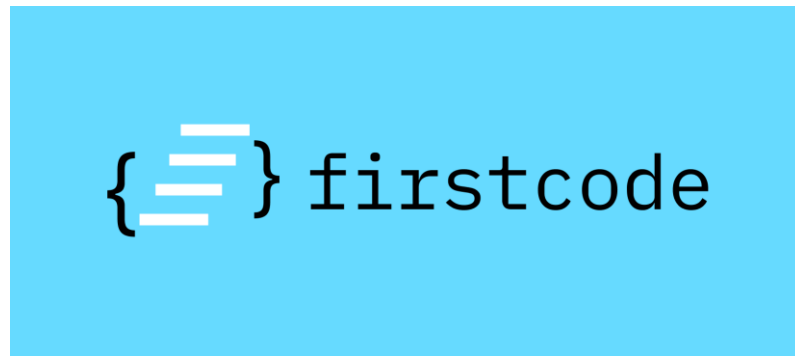


Рисунок 6.5 – Логотип проектного сайту

На сайті зустрічаються мінімалістичні векторні іконки, зокрема на Головній сторінці у блоці із переліком усіх уроків для полегшення сприйняття текстової інформації. В блоках із теоретичною частиною присутні декоративні маркери списку для більшої виразності контенту, за допомогою яких акцентується увага користувача.

На проектованому сайті реалізована мінімальна анімація. При наведенні на посилання для переходу між сторінками змінюється їх колір. На Головній сторінці у секції з переліком уроків, наводячи на текстовий блок, змінюється його колір, під ним утворюється невелика тінь та він плавно підіймається на декілька пікселів по осі Y.

На рисунках 6.6-6.9 наведені приклади частин сторінок проектного веб-сайту із застосованим до них дизайном.

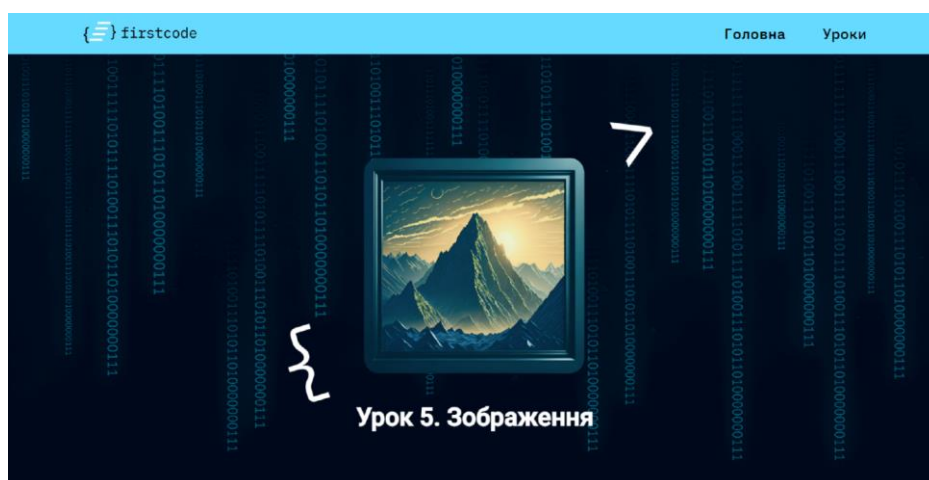


Рисунок 6.6 – Дизайн першого екрану сторінки з уроком по HTML

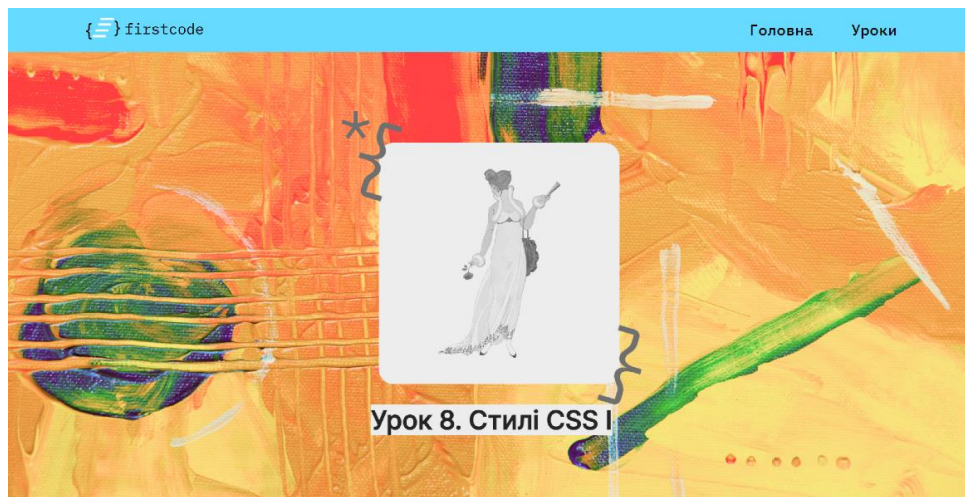


Рисунок 6.7 – Дизайн першого екрану сторінки з уроком по CSS

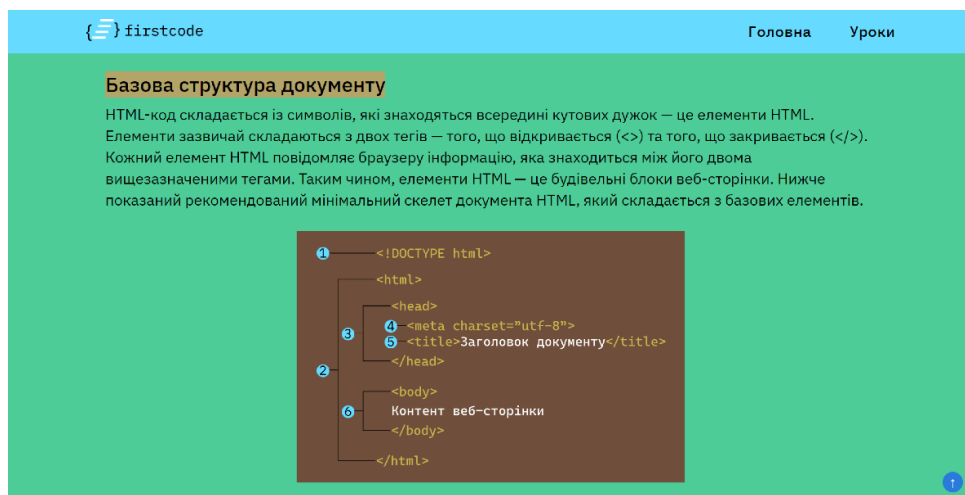


Рисунок 6.8 – Дизайн теоретичного блоку уроку по HTML

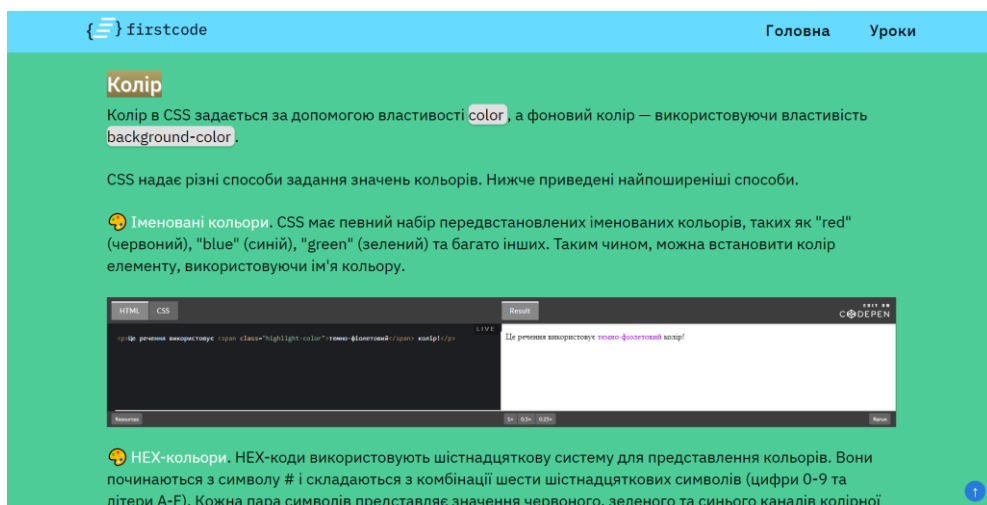


Рисунок 6.9 – Дизайн теоретичного блоку уроку по CSS

7 НАПОВНЕННЯ КОНТЕНТОМ СТОРІНОК ВИДАННЯ

Контент сайту – це інформація, заради якої користувач приходить на сайт. Вміст його сторінок повинен бути структурованим, зрозумілим та відповідати темі сайту, щоб відвідувач на ньому залишився і дійсно знайшов для себе щось корисне.

Інформація, що знаходиться на сайті є авторським трактуванням стандартної інформації стосовно HTML та CSS. Інформація на проєктованому сайті подається лаконічно та структуровано. Вона розбивається на невеликі абзаци, щоб користувачу було легше її сприймати.

Як було зазначено в попередніх розділах, сайт містить Головну сторінку та 9 уроків, які, відповідно, знаходяться на 9 інших сторінках.

Контент Головної сторінки сайту містить загальну інформацію про мови розробки сайту – HTML та CSS, секцію із переліком усіх дев'яти уроків та секцію із зображенням екранів сайту, який буде створений під час уроків.

Уроки розподілені на теоретичні та практичні блоки. В теоретичних блоках йдуть заголовки, підзаголовки загальної теми уроку та основні дані. Кожна теоретична частина уроку по HTML містить базову необхідну інформацію про певну складову веб-сторінки. Так, наприклад, в уроці про таблиці розглянуті всі основні теги, які можуть стати в нагоді при їх будівництві. Якщо уроки стосовно HTML займають 7 сторінок, уроки про стилі CSS – 2 сторінки. В першому уроці щодо стилей йде мова про основні селектори, в другому – демонструються усі базові властивості стилів. Деякі блоки із теорією супроводжуються зображеннями для кращого розуміння, а також прикладами окремого коду. Також в певних частинах присутні таблиці. Після вивчення даних теоретичної частини не буде труднощів із розглядом та розумінням коду сайту в блоці із практикою. В блоці із практичною частиною в кожному уроці додається нова частина коду. Таким чином поступово будується сайт та акцентується увага на тому, що саме було додане до коду.

Усі коди на сайті демонструються за допомогою вбудованого коду з веб-сервісу CodePen. Приклади сторінок із контентом приведені на рис. 7.1-7.6.



Рисунок 7.1 – Приклад контенту Головної сторінки



Рисунок 7.2 – Приклад блоку контенту із теорією

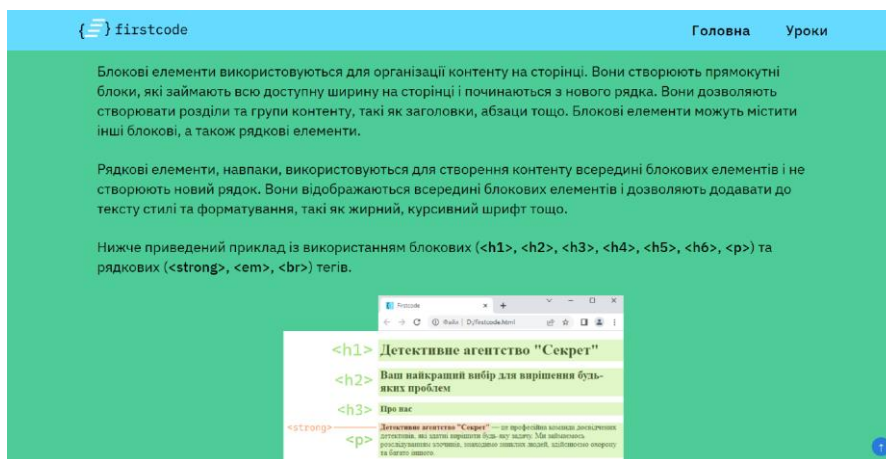


Рисунок 7.3 – Приклад блоку контенту із теорією

Назва селектора	Визначення	Приклад використання
Селектор елемента	Відповідає іменам елементів	<code>p {}</code> Звертається до р елемента
Селектор класу	Відповідає елементу, атрибут класу якого має значення, яке співпадає із значенням, вказаним після символу точки	<code>.container {}</code> Звертається до будь-якого елементу, атрибут класу якого має значення container
Селектор ідентифікатора	Відповідає елементу, чий атрибут id має значення, яке відповідає тому, що вказується після знаку "решітки" (#)	<code>#myElement {}</code> Звертається до елемента, чий атрибут id має значення myElement
Селектор атрибуту	Відповідає елементу, чий атрибут має значення, що відповідає певному критерію або властивості	<code>input[name="username"] {}</code> Звертається до елемента, чий атрибут name має значення username.

Рисунок 7.4 – Приклад блоку контенту із теорією

Нарешті практика

В прикладі використані усі елементи, які були розглянуті в блоці із [теорією](#). За винятком елементів h4-h6, які, як було зазначено, зазвичай в повному складі на веб-сторінці не застосовуються.

В прикладі сайт розподілений на семантичні блоки – із навігацією, заголовком, основною секцією з контентом та підвалом. Основна секція з контентом містить блоки "Послуги", "Про нас" та "Зв'язок з нами". В блоці із переліком послуг для перенесення кожної послуги на новий рядок використовується елемент `br`. В наступному уроці він буде замінений на елементи списку.

В блоці із заголовком сайту продемонстроване використання елементів `strong`, `em` та `span`. До елемента `span` застосований певний стиль: в ньому присутній атрибут `style` із властивістю, яка зменшує жирність шрифту. Детальніше про застосування стилів на сайті буде розглянуто в одному з наступних уроців.

Рисунок 7.5 – Приклад блоку контенту із практикою

Застосування CSS3

Заключним кроком оформлення веб-сайту стане додавання відносно нових властивостей, які прийшли в CSS3. Додаємо тіні до заголовків секцій, іконок та до певних елементів: навігаційного блоку, підвалу, таблиці та зображення. Округлюємо кути у контейнерів секцій, зображення, елементів вводу, кнопки "Надіслати" та радіокнопку. Встановлюємо переходи для елементів навігації, посилань в таблиці, кнопок форми. Залиште відкритим тільки вікно Result, щоб переглянути сайт при достатній ширині екрану.

Рисунок 7.6 – Приклад блоку контенту із практикою

8 ТЕСТУВАННЯ ТА ПУБЛІКАЦІЯ

Важливим етапом в розробці веб-видання є його тестування. До проєктованого веб-сайту було застосовано два суттєвих типи тестувань – на кросбраузерність та на адаптивність. Під час кросбраузерного тестування, відповідно, перевіряється його робота в різних браузерах. Це тестування є важливим, адже кожен веб-браузер може мати свої особливості та різні реалізації стандартів веб-технологій. Проведення такого виду тестування допомагає виявити можливі проблеми з відображенням, взаємодією та функціональністю веб-сайту в різних браузерах. Для тестування було обрано чотири найпоширеніші браузери: Google Chrome, Mozilla Firefox, Opera, Microsoft Edge. Під час тестування сайт відображався однаково в усіх чотирьох, не було помічено якихось відмінностей і у функціоналі. Таким чином, тестування на кросбраузерність пройшло успішно. Скриншоти із відображенням сайту у чотирьох браузерах приведені на рисунках 8.1-8.8.

Також було проведено тестування на адаптивність. В наш час, коли користувач має доступ до сайту з різних пристроїв, важливо адаптувати свій сайт до різних екранів, щоб не втратити певну частку аудиторії. При тестуванні веб-сайту на адаптивність необхідно переконатися, що контент, макет та функціонал веб-сайту пристосовуються до різних розмірів екранів, зокрема екранів комп'ютеру, планшету та телефону, і забезпечують зручну взаємодію для всіх користувачів. Для адаптивності проєктованого веб-сайту в CSS-коді були задіяні `media queries`, які дозволяють налаштовувати відображення та орієнтацію контенту при різних розмірах екрану. Тестування проводилося за допомогою інструментів розробника DevTools браузера Chrome. В них присутня функція `Dimensions`, за допомогою якої можна обрати необхідну модель пристрою, або задати роздільну здатність екрану вручну. Тестування проводилося на екранах телефону, планшету та комп'ютеру, під час якого екран проєктованого сайту успішно адаптувався.

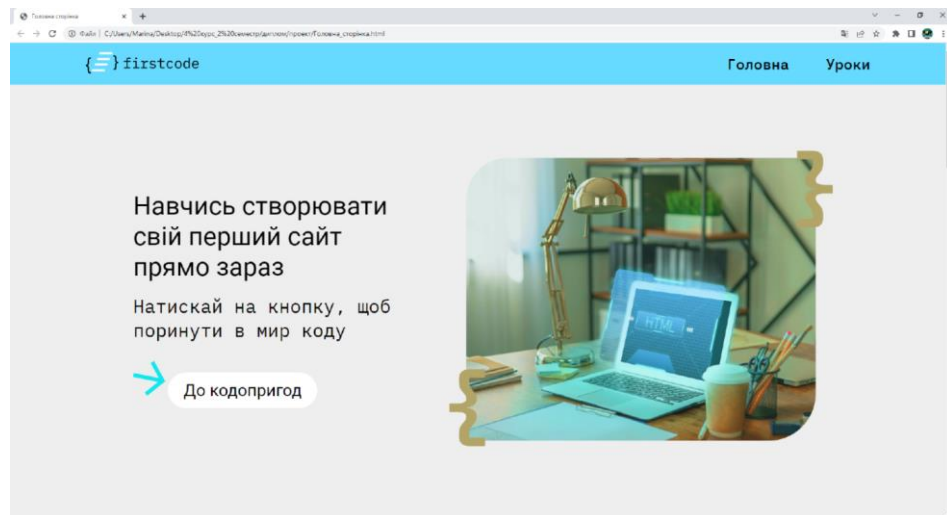


Рисунок 8.1 – Відображення Головної сторінки в Google Chrome

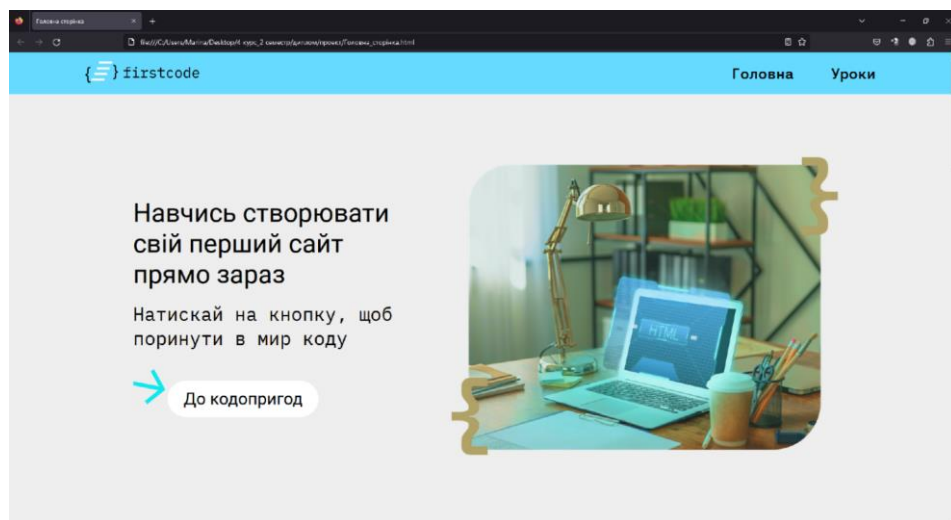


Рисунок 8.2 – Відображення Головної сторінки в Mozilla Firefox

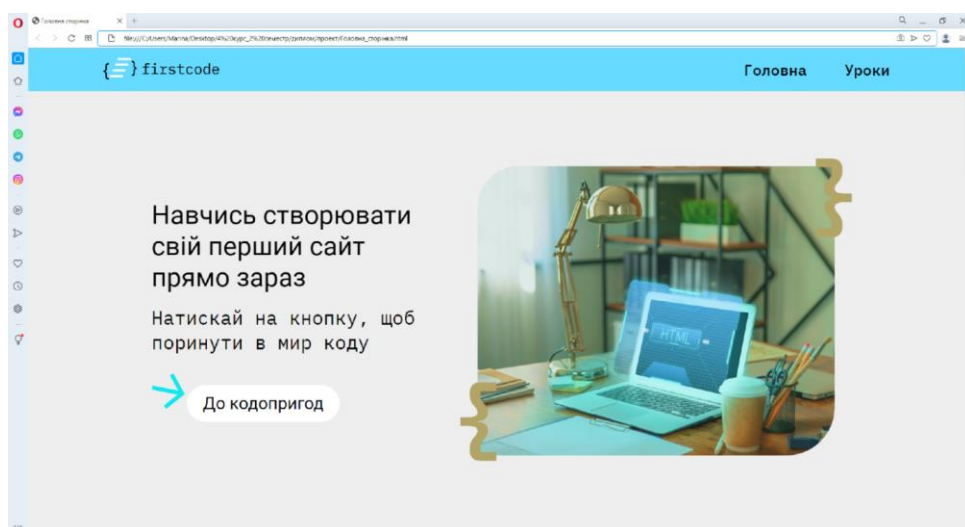


Рисунок 8.3 – Відображення Головної сторінки в Opera

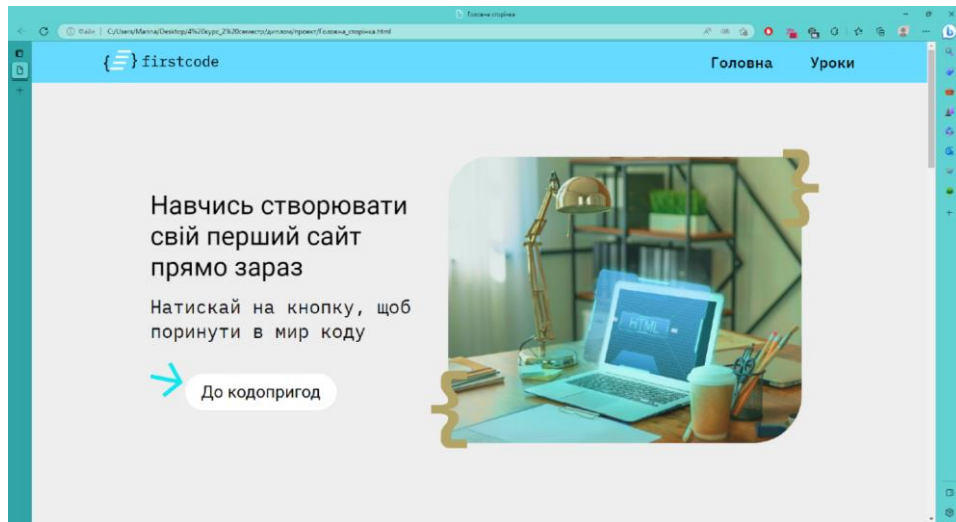


Рисунок 8.4 – Відображення Головної сторінки в Microsoft Edge

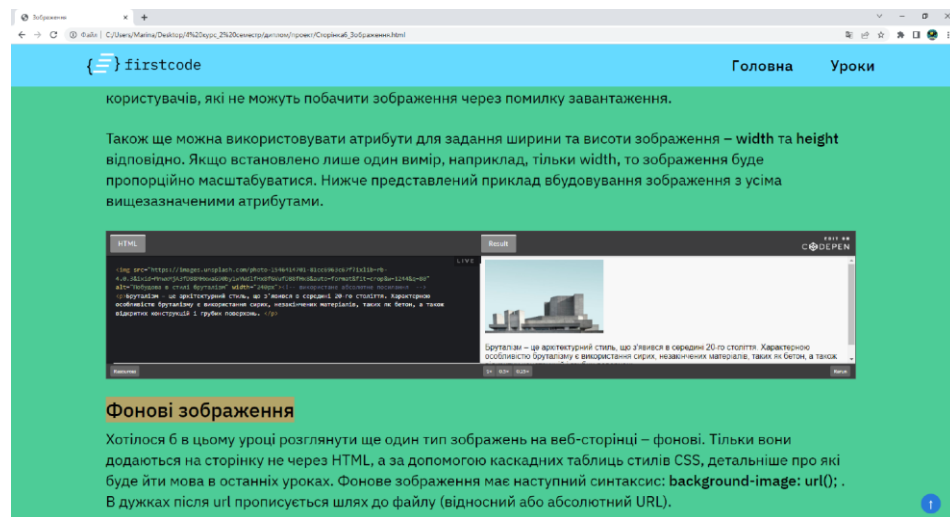


Рисунок 8.5 – Відображення блоку уроку в Google Chrome

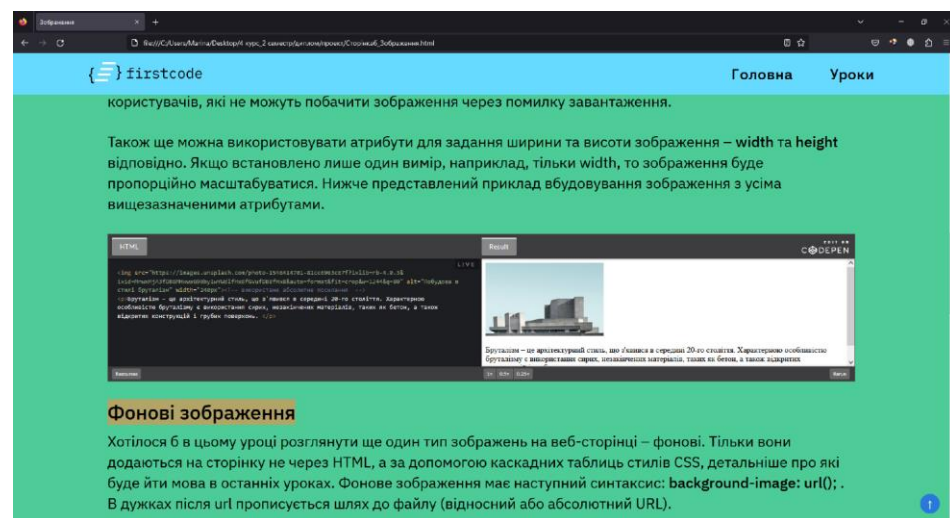


Рисунок 8.6 – Відображення блоку уроку в Mozilla Firefox

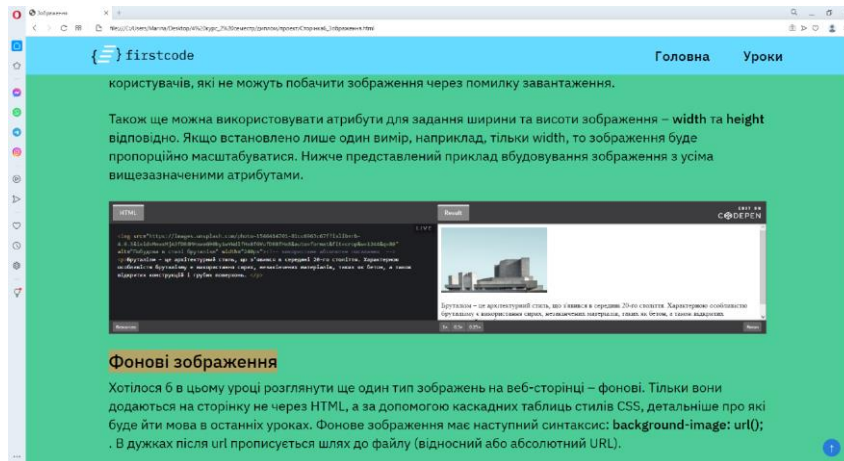


Рисунок 8.7 – Відображення блоку уроку в Орега

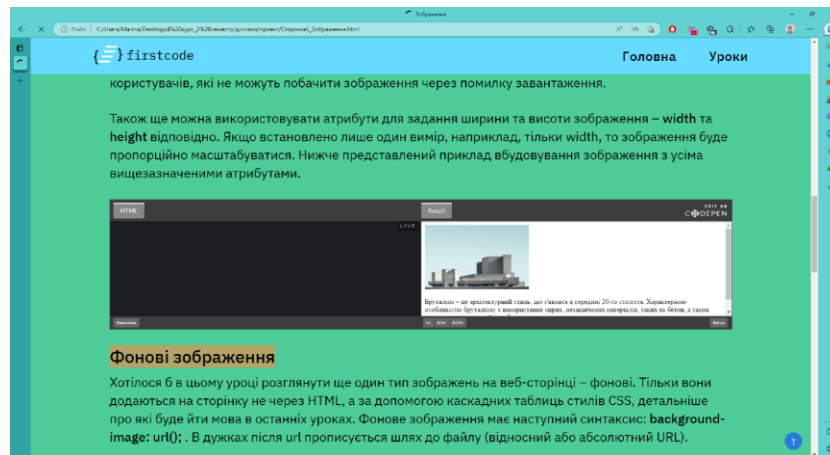


Рисунок 8.8 – Відображення блоку уроку в Microsoft Edge

Скриншоти екранів трьох пристроїв з відображенням сайту приведені на рисунках 8.9-8.14.

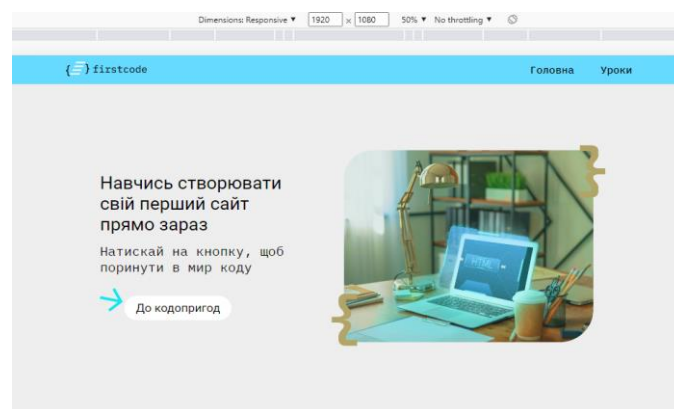


Рисунок 8.9 – Відображення сайту на екрані ноутбуку розмірів 1920x1080

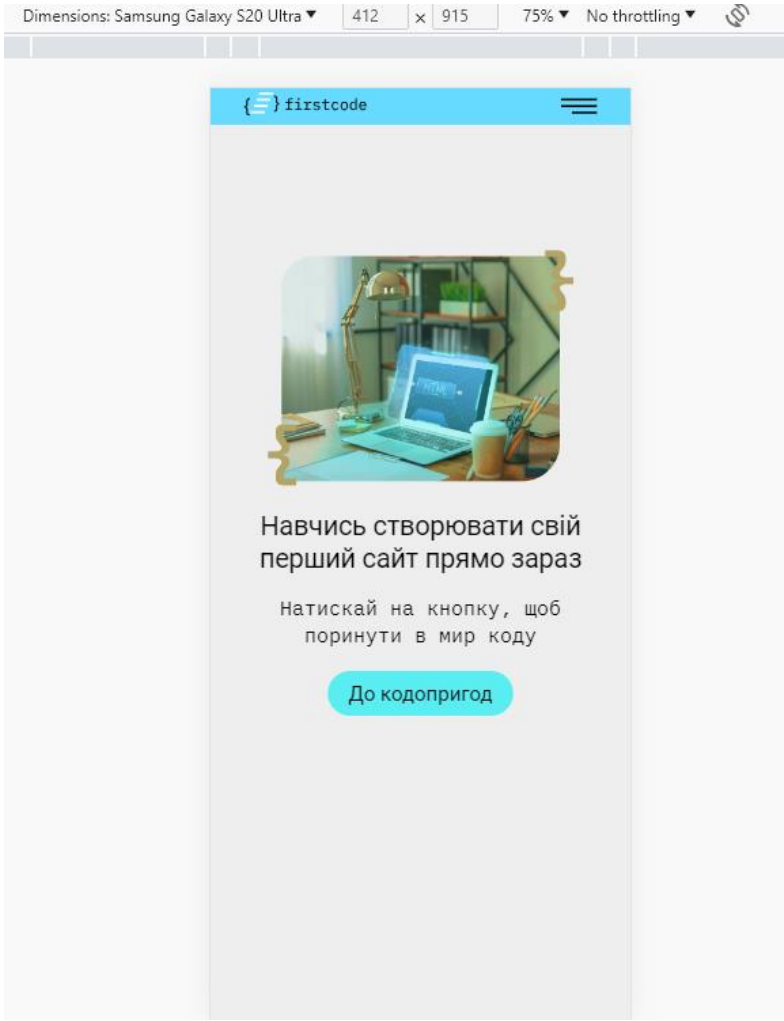


Рисунок 8.10 – Відображення сайту на екрані телефону Galaxy S20 Ultra розмірів 412x915

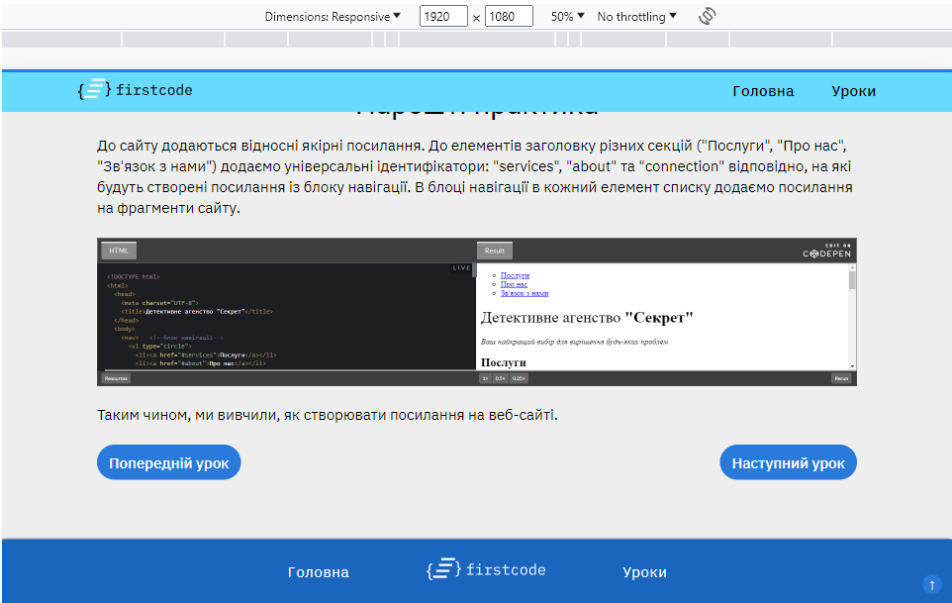


Рисунок 8.11 – Відображення сайту на екрані ноутбуку розмірів 1920x1080

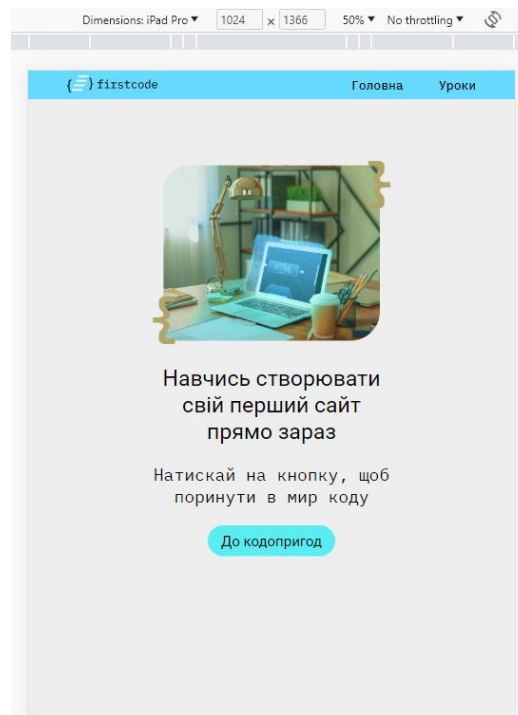


Рисунок 8.12 – Відображення сайту на екрані планшету iPad Pro розмірів 1024x1366

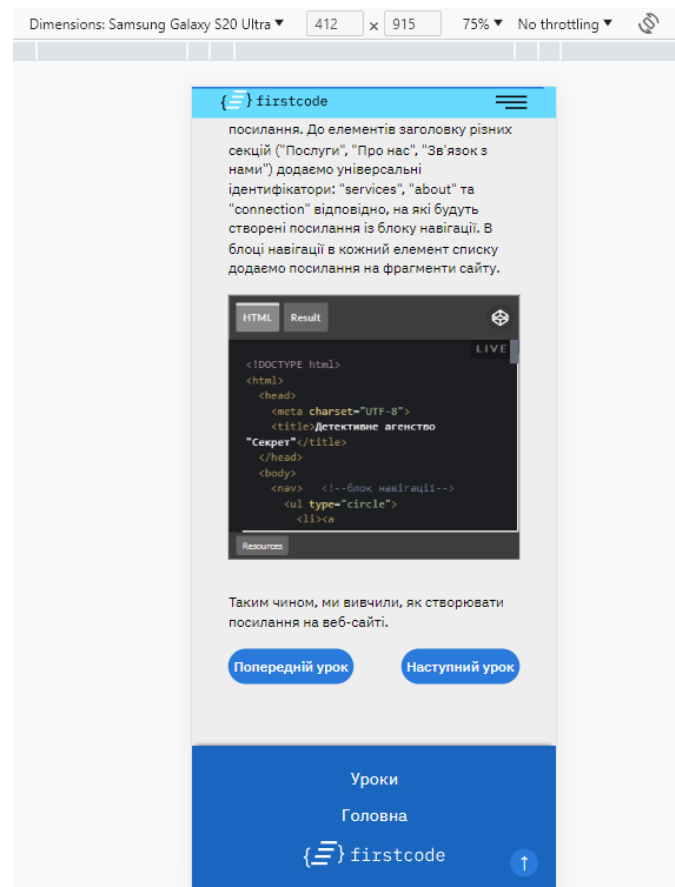


Рисунок 8.13 – Відображення сайту на екрані телефону Galaxy S20 Ultra розмірів 412x915

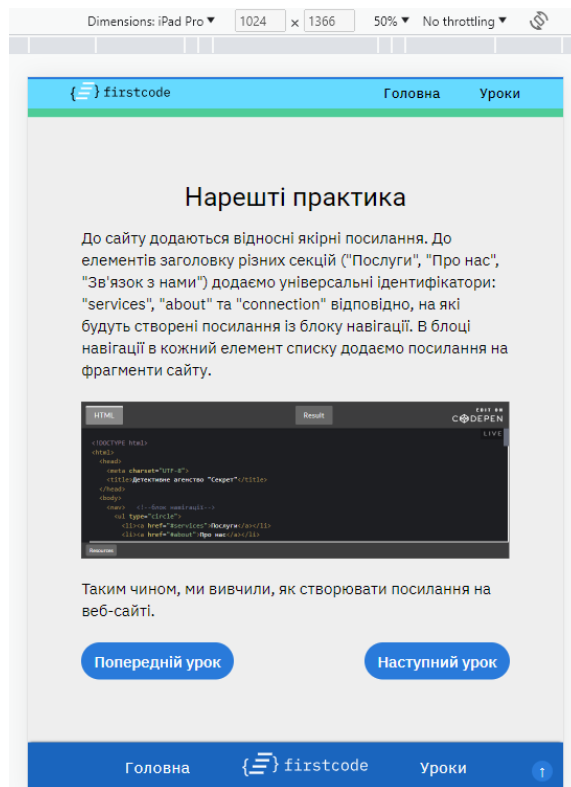


Рисунок 8.14 – Відображення сайту на екрані планшету iPad Pro розмірів 1024x1366

Після проведення успішного тестування, сайт був опублікований на безкоштовному хостингу Neocities. Скриншот екрану із відображенням завантажених файлів на хостинг представлений на рисунку 8.15.

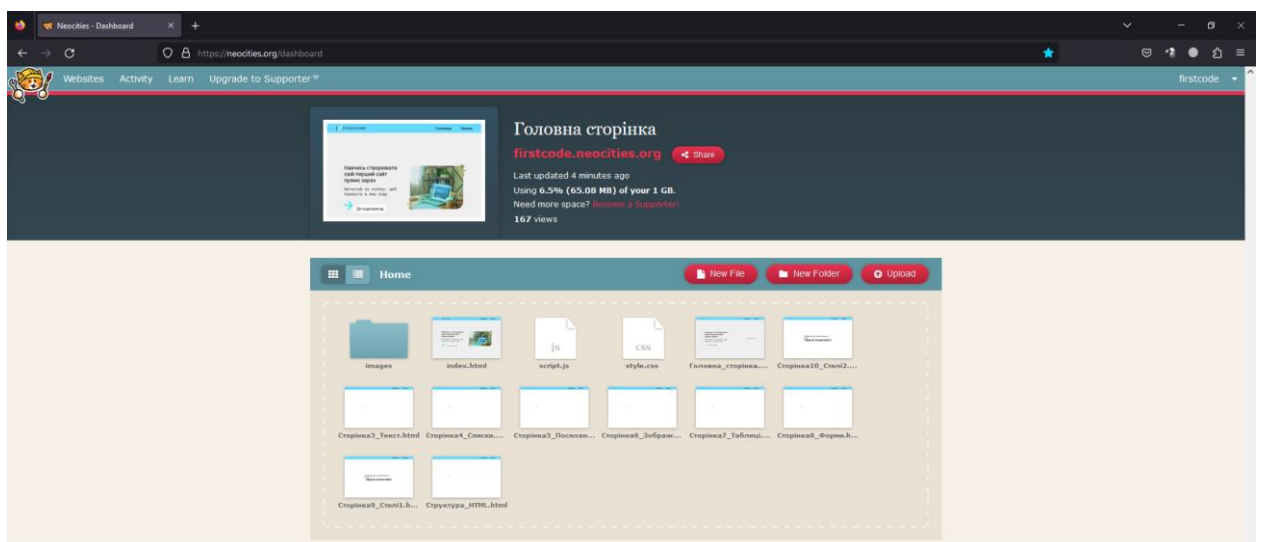


Рисунок 8.15 – Завантажені файли на хостинг

9 ЕКОНОМІЧНА ЧАСТИНА

В результаті виконання кваліфікаційної роботи розроблений навчальний web-сайт «Уроки створення сайту засобами HTML5 та CSS3». Метою сайту є формування певної бази знань початківців у веб-розробці.

Проаналізувавши потенційних конкурентів, було з'ясовано, що сайтів відповідної тематики саме з уроками створення сайтів засобами HTML5 та CSS3, зовсім небагато. Але попит на сайти з такою спрямованістю великий, адже сфера ІТ в наш час є достатньо популярною, що підкреслює актуальність розробленого сайту.

Аналіз аналогічних сайтів показав, що на більшості з них інформація структурована незручним чином, а також присутній непривабливий, застарілий дизайн, що, звичайно, може відштовхнути користувача від вивчення необхідної теми. Серед переваг розробленого сайту, в порівнянні з його конкурентами, можна виділити зручно структуровану та лаконічно подану інформацію; зручну навігацію, за допомогою якої орієнтація на сайті є швидкою та легкою; сучасний дизайн; наявність певної інтерактивності у вигляді вбудованого коду, який можна редагувати та відразу бачити його результат.

Веб-сайт є статичним, інформація на ньому є релевантною, що не потребує регулярного оновлення. Тому, в порівнянні з динамічним типом сайту, розроблений сайт потребує менших фінансових витрат, що можна вважати його безумовною перевагою.

Крім того, кожна сторінка сайту, за винятком Головної, має однакову структуру, що полегшує процес створення дизайну та верстання сайту та значно скорочує час розробки. Також на час розробки впливає те, що ілюстрований контент сайту по кількості дуже помірний, таким чином, на підбір та графічну обробку такого типу контенту була витрачена невелика кількість часу.

Завдяки компактному обсягу файлів за кількістю мегабайтів, з яких складається сайт, та використанням стандартних технологій для його розробки, не виникне проблем з його розміщенням на безкоштовному хостингу, що також зменшує витрати.

Важливим етапом в економічному обґрунтуванні проекту є визначення джерел витрат, економії, фінансування та прибутку.

Планується, що веб-сайт буде використовуватись в рамках роботи школи з навчання веб-розробці. Витрати на розробку веб-сайту здійснюються школою, студентам якої буде наданий доступ до створеного сайту.

Витрати на створення веб-сайту є незначними з огляду на те, що сайт статичний, кожна сторінка містить однакову структуру, що значно скорочує час розробки. Сайт також буде розміщений на безкоштовному хостингу, що не потягне за собою додаткових витрат.

Фінансує проект школа з навчання веб-розробки, студенти якої за додаткову плату будуть мати доступ до розробленого сайту. Відповідно, прибуток буде формуватися за рахунок цих коштів.

Для розрахунку витрат на заробітну плату необхідно конкретизувати етапи створення веб-сайту. Процес розробки веб-сайту «Уроки створення сайту засобами HTML5 та CSS3» містить наступні етапи:

- аналіз ТЗ, який включає в себе визначення всіх вимог та деталей проекту;
- проектування інформаційної структури та навігації;
- макетування сторінок, під час якого створюється дизайн сайту;
- створення контенту, що включає в себе написання та редагування тексту;
- верстка сторінок, під час якої створений макет перетворюється у код;
- тестування та публікація сайту.

Наступним кроком є розрахунок собівартості та ціни розробки сайту. До собівартості розробки сайту відносяться такі статті витрат: основна заробітна плата; додаткова заробітна плата; єдиний соціальний внесок; інші витрати.

В процесі розробки веб-сайту задіяні наступні виконавці: менеджер, веб-дизайнер, копірайтер та веб-програміст. Заробітна плата менеджера проекту в місяць становить 20 000 грн; веб-дизайнера – 23 000 грн; копірайтера – 17 600 грн; веб-програміста – 30 000 грн. Робочий день триває 8 годин. Кількість днів для розробки сайту – 20.

Кількість робочих днів в місяць приблизно складає 22 дні. Враховуючи, що робочий день триває 8 годин, розрахована кількість робочих годин в місяці складає 176.

Потім знаходиться годинна ставка для кожної посади:

- менеджер проекту: $20\,000 / 176 = 114$ грн/год.;
- веб-дизайнер: $23\,000 / 176 = 131$ грн/год.;
- копірайтер: $17\,600 / 176 = 100$ грн/год.;
- веб-програміст: $30\,000 / 176 = 170$ грн/год.

Виходячи з тривалості роботи фахівців та годинної ставки, розраховується заробітна плата для кожної посади:

- менеджер проекту: $2 * 8 = 16$; $16 * 114 = 1\,824$ грн;
- веб-дизайнер: $5 * 8 = 40$; $40 * 131 = 5\,240$ грн;
- копірайтер: $6 * 8 = 48$; $48 * 100 = 4\,800$ грн;
- веб-програміст: $7 * 8 = 56$; $56 * 170 = 9\,520$ грн.

Додаткова заробітна плата – це винагорода за працю понад установлені норми, за трудові успіхи й винахідливість, за особливі умови праці. Включає доплати, надбавки, гарантійні та компенсаційні виплати, передбачені чинним законодавством, премії, пов'язані з виконанням робочих завдань [23].

В рамках даної роботи було прийнято, що додаткова заробітна плата становить 20 % від основної:

$$21\,384,00 * 0,2 = 4\,276,80 \text{ грн.}$$

Результати розрахунку заробітної плати виконавців роботи приведені в таблицях 9.1.

Таблиця 9.1 – Результати розрахунку заробітної плати виконавців

Етап	Вид робіт	Виконавець		Годинна ставка, грн	Тривалість виконання, дні	Заробітна плата, грн
		Кількість	Посада			
1. Аналіз ТЗ	Визначення вимог до сайту	1	менеджер проекту	114,00	1	912,00
2. Організація структури сайту	Проектування інформаційної структури та навігації	1	веб-дизайнер	131,00	2	2 096,00
3. Макетування	Розробка дизайну	1	веб-дизайнер	131,00	3	3 144,00
4. Створення контенту	Написання та редагування тексту	1	копірайтер	100,00	6	4 800,00
5. Верстка	Написання коду HTML, CSS, JS	1	веб-програміст	170,00	6	8 160,00
6. Тестування		1	веб-програміст	170,00	1	1 360,00
7. Публікація		1	менеджер проекту	114,00	1	912,00
Разом					20	21 384,00
Додаткова заробітна плата (20 %)						4 276,80
Усього						25 660,80

Єдиний соціальний внесок становить 22 % від величини основної та додаткової заробітної плати:

$$(21\,384,00 + 4\,276,80) * 0,22 = 5\,645,38 \text{ грн.}$$

Також враховані витрати на електроенергію та обслуговування комп'ютерів, які були віднесені до інших витрат.

Витрати на електроенергію розраховуються шляхом добутку споживної потужності, тарифу на електроенергію та кількості годин роботи обладнання. При розробці сайту використано 2 ноутбуки, які мають потужність 0,054 кВт/год. Приймається, що вартість 1 кВт/год становить 1,68 грн. Під час розробки сайту електроенергія споживається наступну кількість годин:

$$20 * 8 = 160 \text{ год.}$$

Таким чином, витрати на електроенергію дорівнюють:

$$0,054 * 1,68 * 160 * 2 = 29,03 \text{ грн.}$$

Для розрахунку витрат на обслуговування ЕОМ береться її вартість та час експлуатації (як правило, не більше 3 років). ЕОМ за рік використовується 254 робочих дні. Таким чином, витрати на обслуговування ЕОМ дорівнюють:

$$20\,000 / (3 * 8 * 254) * 160 = 525 \text{ грн.}$$

Собівартість створеного сайту складає:

$$21\,384,00 + 4\,276,80 + 5\,645,38 + 525,00 + 29,03 = 31\,860,21 \text{ грн.}$$

Розраховується сума прибутку, прийнявши рентабельність у 30 %:

$$31\,860,21 * 0,3 = 9\,558,06 \text{ грн.}$$

Розраховується ціна розробки без податку на додану вартість (ПДВ):

$$31\,860,21 + 9\,558,06 = 41\,418,27 \text{ грн.}$$

Далі визначається сума ПДВ, яка складає 20 % від ціни:

$$41\,418,27 * 0,2 = 8\,283,65 \text{ грн.}$$

Наприкінці розраховується ціна розробки сайту з урахуванням ПДВ:

$$41\,418,27 + 8\,283,65 = 49\,701,92 \text{ грн.}$$

Результати розрахунків занесені в таблицю 9.2.

Таблиця 9.2 – Результати розрахунку витрат на розробку сайту

Стаття витрат	Сума, грн
Основна заробітна плата	21 384,00
Додаткова заробітна плата	4 276,80
Єдиний соціальний внесок	5 645,38
Витрати на обслуговування ЕОМ	525,00
Витрати на електроенергію	29,03
Собівартість розробки сайту	31 860,21
Прибуток	9 558,06
Ціна без ПДВ	41 418,27
Податок на додану вартість (ПДВ)	8 283,65
Ціна з урахуванням ПДВ	49 701,92

Таким чином, повна ціна розробки веб-сайту становить 49 701,92 грн. При розробці веб-сайту задіяні: менеджер проекту, веб-дизайнер, копірайтер та веб-програміст. Кількість днів на розробку – 20. Очікувана сума прибутку складатиме 9 558,06 грн.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи бакалавра створено адаптивний web-сайт на тему «Уроки створення сайту засобами HTML5 та CSS3».

Під час розроблення сайту було виконано ряд завдань, а саме: вибір інструментальних засобів для розроблення сайту, проектування інформаційної структури та навігації, розробка модульної сітки, створення графічного дизайну, наповнення сторінок веб-видання контентом, тестування сайту на кросбраузерність та адаптивність, публікація сайту.

Спроектований сайт має освітню спрямованість, зайшовши на який можна сформуванати базу знань стосовно мов HTML та CSS. Сайт має Головну сторінку та дев'ять сторінок із уроками. Кожний урок містить теоретичний блок, отримавши знання з якого, можна переходити до практичного блоку та слідкувати в ньому за створенням коду для сайту. Із кожним уроком код, відповідно, стає все більше наповненим. Демонстрований код є вбудованим за допомогою платформи CodePen. Він є інтерактивним: користувачі можуть маніпулювати кодом та у правому віконці бачити результат коду. Також на сайті застосована зручна навігація у вигляді кнопок для переходу до попереднього та наступного уроків, а також загального блоку з гіперпосиланнями до кожного уроку.

Розробку сайту здійснено з використанням мов HTML, CSS та JavaScript та програми Visual Studio Code. Для створення графічного дизайну сайту були використані такі програмні засоби, як Figma та Adobe Photoshop.

Результатом виконання кваліфікаційної роботи бакалавра є адаптивний кросбраузерний web-сайт на тему «Уроки створення сайту засобами HTML5 та CSS3».

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The best place to build, test, and discover front-end code. URL: <https://codepen.io/> (дата звернення: 23.05.2023).
2. Carey, P. M. New Perspectives HTML5 and CSS3: Comprehensive. Cengage Learning, 2017. 872 p.
3. Gasston, P. The Modern Web: Multi-Device Web Development with HTML5, CSS3, and JavaScript. No Starch Press, 2013. 264 p.
4. Єгорова І.М., Панасенко В.Л. О сферах використання сучасних стандартів верстки веб-документів // Поліграфічні, мультимедійні та web-технології: тези доп. IV Міжнар. наук.-техн. конф., 14-17 травня 2019 р. Харків: «Друкарня Мадрид», 2019. Т. 1. С. 133-134.
5. McFarland D.S. JavaScript & jQuery: The Missing Manual. O'Reilly Media, 2014. 686 p.
6. Єгорова І.М. WEB-технології: методичні вказівки до самостійної роботи. Харків: ХНУРЕ, 2017. 90 с.
7. Visual Studio Code. URL: https://en.wikipedia.org/wiki/Visual_Studio_Code(дата звернення: 23.05.2023).
8. Code editing. Redefined. URL: <https://code.visualstudio.com/> (дата звернення: 23.05.2023).
9. IntelliSense. URL: <https://code.visualstudio.com/docs/editor/intellisense> (дата звернення: 23.05.2023).
10. It's time to design on your terms. URL: <https://www.figma.com/design/> (дата звернення: 23.05.2023).
11. Start with Photoshop. Amazing will follow. URL: <https://www.adobe.com/products/photoshop.html> (дата звернення: 23.05.2023).
12. Gorgeous graphics, designed by you. URL: <https://www.adobe.com/products/illustrator.html> (дата звернення: 23.05.2023).
13. What is Responsive Design? URL: <https://www.browserstack.com/responsive-design> (дата звернення: 23.05.2023).
14. How to Perform Responsive Testing for a Locally Hosted Website. URL: <https://www.browserstack.com/guide/responsive-testing-on-local-server> (дата звернення: 23.05.2023).

15. What is cross-browser support? URL: <https://www.g2.com/glossary/cross-browser-support-definition> (дата звернення: 23.05.2023).
16. Deineko Zh., Zeleniy O., Lyashenko V., Tabakova I. (2021). Color space image as a factor in the choice of its processing technology // Problems of modern science and practice (September 21-24, 2021). Boston, USA. P. 389-394.
17. Методичні вказівки з виконання кваліфікаційної роботи для студентів денної та заочної форми навчання першого (бакалаврського) рівня вищої освіти спеціальності 186 «Видавництво та поліграфія» за освітньою програмою «Видавничо-поліграфічна справа» / В.П. Ткаченко, А.В. Бізюк, О.В. Вовк, І.М. Єгорова, В.Ф. Челомбійко. Харків: ХНУРЕ, 2020. 68 с.
18. Єгорова І.М., Короткевіч Ю.В. Щодо створення графічного інтерфейсу сайтів // Радіоелектроніка та молодь у ХХІ столітті: зб. матеріалів 25-го Міжнар. молодіжн. форуму, 20–22 квітня 2021 р. Харків: ХНУРЕ, 2021. Т. 6. С. 389–390.
19. What is the website grid? URL: <https://www.impactplus.com/blog/what-is-the-website-grid> (дата звернення: 27.05.2023).
20. Is Color Preference Affected by Age Difference – Scientific Figure on ResearchGate. URL: https://www.researchgate.net/figure/The-color-preference-order-for-different-age-groups_fig1_267857549 (дата звернення: 28.05.2023).
21. Колесникова Т.А. Основи теорії кольору та кольоровідтворення: конспект лекцій. Харків: ХНУРЕ, 2017. 68 с.
22. Вовк О.В., Шипова М.К. Психоемоційний вплив кольору // Поліграфічні, мультимедійні та web-технології: матеріали Молодіжної школи-семінару V Міжнародної науково-технічної конференції, 3 листопада 2020 р. Харків: ХНУРЕ, 2020. Т. 2. С. 11-13.
23. Полозова Т.В. Конспект лекцій з дисципліни «Економіка та бізнес» для здобувачів усіх форм навчання першого (бакалаврського) рівня вищої освіти спеціальності 186 «Видавництво та поліграфія» освітньо-професійної програми «Видавничо-поліграфічна справа». Харків: ХНУРЕ, 2020. 111 с.