



# Звіт про оригінальність

● Оцінка схожості

% 7

● Ризик плагіату

СЕРЕДНІЙ

👤 Ігор Кагало 🕒 2025-06-14 10:19

Посилання на звіт: 10bie / Посилання користувача: qAHy



# Ось вона – Ваша звіт про оригінальність!

Ми раді повідомити, що перевірка вашого документа завершена, і результати вже готові! Наші алгоритми старанно працювали, щоб знайти збіги в наших базах даних.

На наступних сторінках ви знайдете результати перевірки:

---

Бали

---

Збіги

---

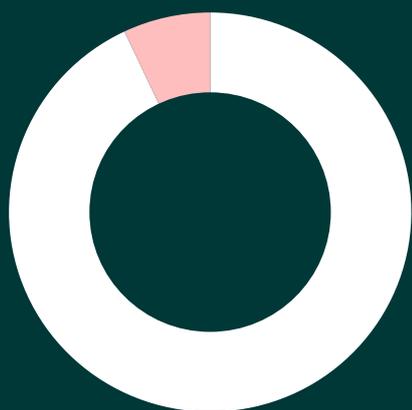
Посилання

---

Ваш документ було перевірено за такими джерелами:

- База даних інтернет-джерел
- База даних наукових статей
- Глибока перевірка (наш вдосконалений алгоритм)

# Бали



● Збіги тексту	7%
● Перефразування	0%
● Цитований текст	0%
● Неправильне цитування	0%
● Збігів не знайдено	93%

## Ризик плагіату

**СЕРЕДНІЙ**

Ризик плагіату вказує, як збіги тексту розподілені по документу. Вищий ризик виникає, коли збіги з'являються близько один до одного, наприклад, у тому самому абзаці або розділі.

## Оцінка схожості

Оцінка схожості показує, скільки слів або символів у вашому документі збігаються з текстами інших документів, включаючи перефразовані тексти або неправильні цитати.

% **7**

# Збіги

---

## 1 ОГЛЯД СУЧАСНИХ ТЕНДЕНЦІЙ У РЕКЛАМІ ЦИФРОВОЇ ТЕХНІКИ

### Значення веб-блогів у рекламній діяльності

У сучасному цифровому середовищі веб-блог стає потужним інструментом для просування продукції, зокрема, цифрової техніки. З розвитком Інтернету та інтерактивних платформ, веб-блог забезпечує брендам можливість **41** безпосередньо взаємодіяти з потенційними покупцями, надаючи детальну інформацію про характеристики продукції, інструкції та відгуки, що є критичними для прийняття рішення про покупку. Особливо це важливо у сфері цифрової техніки, де конкуренція велика, а споживачі часто шукають об'єктивні огляди, порівняння та поради щодо вибору оптимальних товарів.

Для того, щоб веб-блог став ефективним інструментом реклами, необхідно створити інтерактивний контент, який буде залучати відвідувачів та допомагати їм приймати обґрунтовані рішення. Це передбачає використання новітніх веб-технологій та інструментів розробки, таких як VisualStudioCode (VS Code), що дає можливість легко інтегрувати різноманітні інтерактивні елементи без необхідності писати складний код вручну.

### Основні переваги веб-блогу для реклами цифрової техніки:

**Інтерактивний контент.** Веб-блог може містити інтерактивні елементи, такі як слайдери для порівняння технічних характеристик різних моделей, відеоогляди продуктів, інтерактивні карти точок продажу, а також форми зворотного зв'язку для збору контактної інформації. Ці елементи сприяють більш глибокому залученню користувачів до контенту.

**Візуальний контент.** Створення привабливих банерів, інфографіки та рекламних матеріалів за допомогою графічних редакторів, таких як Canva, Figma або AdobePhotoshop, додає блогу професійного вигляду та підвищує його візуальну привабливість. Хороший дизайн значно підвищує ефективність реклами цифрової техніки.

SEO-оптимізація. Веб-блоги є важливим інструментом для покращення позицій сайту в 45 пошукових системах. Використання ключових слів, правильне оформлення контенту та оновлення інформації на блозі допомагають залучати більше органічного трафіку.

Довготривалий ефект. На відміну від традиційної реклами, статті на блозі зберігають свою актуальність протягом тривалого часу, що дозволяє досягти тривалого ефекту в рекламі цифрової техніки.

Залучення аудиторії через коментарі та зворотний зв'язок. Завдяки інтерактивним можливостям веб-блогів, користувачі можуть залишати коментарі, запитання або ділитися відгуками про техніку. Це створює додаткову взаємодію з брендом і дозволяє виробникам отримувати цінний зворотний зв'язок.

Персоналізація контенту. За допомогою інтерактивних елементів можна персоналізувати враження відвідувачів блогу, пропонуючи їм релевантні матеріали на основі їхніх інтересів або поведінки на сайті.

Використання сучасних інструментів розробки веб-сайтів, таких як VisualStudioCode, є ключовим для створення інтерактивного веб-блогу. Це середовище надає всі необхідні інструменти для роботи з веб-технологіями, зокрема HTML, CSS, JavaScript, а також інтеграцію з такими фреймворками, як React або Vue.js, для реалізації динамічних та інтерактивних елементів на сайті.

Крім того, сучасні веб-блоги можуть інтегрувати технології штучного інтелекту для автоматизації взаємодії з користувачами, наприклад, чат-боти для миттєвих консультацій, алгоритми персоналізованих рекомендацій товарів та інструменти аналізу поведінки відвідувачів. Це дозволяє значно підвищити ефективність рекламних кампаній та забезпечити кращий користувацький досвід.

Загалом, роль веб-блогів у рекламній діяльності цифрової техніки важко переоцінити. Вони дають можливість не лише інформувати потенційних клієнтів про новинки на ринку, але й забезпечують активну взаємодію з ними, що дозволяє побудувати лояльність і стимулювати продаж товарної продукції.

Огляд сучасних програмних засобів, їх інструментів та середовищ розробки для створення інтерактивного контенту

Створення інтерактивного веб-блогу для реклами цифрової техніки передбачає використання сучасних веб-технологій та інструментів для створення привабливого і функціонального контенту. Інтерактивність на веб-блогах забезпечує не лише залучення аудиторії, але й підвищує ефективність реклами продукції. Важливим моментом є використання середовища розробки, такого як VisualStudioCode (VS Code),

яке дозволяє зручно працювати з різними мовами програмування та бібліотеками для створення інтерактивних веб-додатків.

Для створення інтерактивного веб-блогу, який має ефективно рекламувати цифрову техніку, необхідно використовувати різні веб-технології, що забезпечують динамічну взаємодію з користувачем та привабливий дизайн.

HTML є основною технологією для створення структури веб-сторінки. Завдяки підтримці мультимедійних елементів (відео, аудіо) і інтерактивних елементів (форми, кнопки, списки) HTML дозволяє реалізовувати функціональність веб-блогу, де користувач може взаємодіяти з контентом. CSS використовується для стилізації веб-сторінки та додавання анімацій, що дозволяє зробити веб-блог більш динамічним і привабливим для користувачів. Для інтерактивних ефектів можна використовувати трансформації, переходи та анімації, що полегшує створення сучасного дизайну блогу для реклами продукції.

JavaScript є основною мовою програмування для створення динамічних і інтерактивних елементів на веб-сторінці. Він дозволяє обробляти події, змінювати вміст сторінки без перезавантаження (через AJAX), а також створювати інтерфейси для взаємодії з користувачем.

Для полегшення роботи з JavaScript та створення інтерактивних елементів використовуються популярні бібліотеки та фреймворки:

jQuery – спрощує роботу з DOM (DocumentObjectModel), дозволяючи швидко реалізувати анімації, ефекти та обробку подій.

React – бібліотека для побудови користувацьких інтерфейсів, яка дозволяє створювати компонентні веб-додатки та забезпечує швидке оновлення контенту без перезавантаження сторінки.

Vue.js – ще один популярний фреймворк для побудови інтерактивних інтерфейсів. Відзначається легкістю у навчанні, водночас забезпечуючи високу гнучкість і ефективність розробки.

Для створення серверної частини веб-блогу можна використовувати:

Node.js – платформу **24** для виконання JavaScript на сервері. Вона дозволяє створювати **масштабовані веб-додатки** з високою продуктивністю.

Express – це веб-фреймворк для Node.js, який полегшує створення серверних додатків, таких як обробка запитів від користувачів, робота з базами даних, а також маршрутизація.

Ці інструменти допомагають створити серверну частину для вашого веб-блогу, що дозволить реалізувати додаткові функціональні можливості, такі як персоналізація контенту або взаємодія з базами даних.

Для зберігання інформації про продукти (цифрову техніку) можна використовувати реляційні або нереляційні бази даних:

MySQL або PostgreSQL є реляційними базами даних, які використовуються для зберігання структурованих даних, таких як інформація про продукти, відгуки користувачів, історія покупок тощо.

MongoDB **21** – це нереляційна база **21** даних, яка **21** дозволяє зберігати дані в форматі JSON, що зручно для веб-додатків з динамічними даними.

### 1.3 Огляд інструментів графічного дизайну для створення інтерактивного **34** контенту

**34** Візуальний контент відіграє важливу роль у привабливості та ефективності реклами на веб-блогах. Інтерактивний контент потребує високоякісних візуальних **4** елементів, таких як банери, інфографіки, ілюстрації, а також інтерфейсні елементи. Для цього існує ряд потужних графічних редакторів, які дозволяють створювати професійні матеріали без необхідності використання складних програм для дизайну.

Canva – це **4** інтуїтивно зрозумілий інструмент для створення графіки, банерів, інфографіки та інших візуальних елементів. Canva особливо корисна для швидкого створення маркетингових матеріалів, зокрема для реклами продукції. Завдяки готовим шаблонам можна швидко створити привабливі банери та зображення для веб-блогу без спеціальних дизайнерських навичок.

Adobe Photoshop – професійний графічний редактор, який дозволяє створювати складні візуальні елементи, такі як рекламні графіки, банери, ретушування зображень тощо. Photoshop є стандартом у сфері дизайну, тому його можна використовувати для розробки високоякісних візуальних матеріалів для реклами цифрової техніки.

Figma – це інструмент для командного дизайну, який дозволяє створювати макети веб-сторінок, інтерфейсів та елементів для інтерактивних проектів. Figma зручна для спільної роботи над дизайном і дає змогу створювати адаптивні елементи, які інтегруються з функціоналом веб-блогу.

### Порівняльний аналіз платформ для розробки веб-блогів

У сучасній веб-розробці існує широкий вибір **4** платформ для створення веб-блогів, які пропонують різний рівень функціональності, інтерактивності та зручності використання. Деякі з них орієнтовані на користувачів **4** без глибоких технічних знань

і забезпечують швидке розгортання блогу за допомогою готових шаблонів і візуальних редакторів. Інші, навпаки, надають розширені можливості для кастомізації, інтеграції власного коду та створення унікальних рішень для інтерактивного контенту.

Вибір платформи залежить від кількох ключових аспектів:

Гнучкість налаштувань – можливість змінювати структуру, дизайн і функціональність блогу відповідно до вимог проекту;

Підтримка інтерактивних елементів – використання анімацій, динамічного контенту, інтеграція мультимедіа та інших засобів залучення користувачів;

Простота використання – доступність для розробників і контент-менеджерів, наявність документації та навчальних матеріалів.

Оптимізація та продуктивність – швидкість завантаження сторінок, відповідність стандартам SEO та адаптивність для мобільних пристроїв.

"У цьому підрозділі детально аналізуються найпопулярніші платформи для створення веб-блогів, зокрема WordPress, Wix, Joomla, а також підхід до розробки власного веб-блогу на основі сучасних веб-технологій у середовищі VisualStudioCode. Особливу увагу буде приділено порівнянню можливостей цих платформ із варіантом, коли веб-блог створюється з нуля **25** за допомогою HTML, CSS, JavaScript та серверних технологій, таких як Node.js та Express.

Платформи для створення веб-блогів. Серед найбільш популярних платформ для розробки веб-блогів можна виділити, WordPress. Найпопулярніша **25** платформа для створення блогів, що підтримує як хмарний сервіс (WordPress.com), так і самостійну установку (WordPress.org).

Переваги платформи:

надає великий вибір готових шаблонів і плагінів;

володіє зручною панель адміністратора для керування контентом;

дозволяє здійснювати SEO-оптимізацію та інтеграцію з платними інструментами.

Недоліки платформи:

менша гнучкість у створенні унікальних інтерактивних елементів без використання плагінів;

деякі розширені функції доступні лише у платних версіях.

Wix - конструктор сайтів із можливістю створення блогів за допомогою drag-and-drop інтерфейсу.

Переваги конструктора:

простота у використанні без необхідності володіння мовами програмування;

має вбудовані маркетингові інструменти.

Недоліки конструктора:

обмежені можливості кастомізації та редагування коду;

платні тарифні плани для розширених можливостей.

Joomla. Гнучка система керування вмістом (CMS), що дозволяє створювати складні веб-ресурси, зокрема блоги.

Переваги системи:

є безкоштовною CMS із відкритим кодом, що дозволяє змінювати її відповідно до власних потреб без обмежень;

підтримує модулі, плагіни, компоненти та шаблони, що **35** дає змогу легко додавати нові функції та змінювати дизайн сайту;

володіє зручним візуальним редактором, який дозволяє публікувати, редагувати та організовувати контент без знання коду.

Недоліки системи:

порівняно з WordPress, Joomla потребує більше часу для освоєння;

при великій кількості плагінів може уповільнювати роботу сайту;

складність інтеграції з сучасними фреймворками.

Ghost – легка та швидка платформа для блогів, орієнтована на контент маркетинг.

Переваги платформи:

володіє мінімалістичним інтерфейсом.

оптимізований для швидкості.

Недоліки платформи:

надає менше можливостей кастомізації порівняно з іншими CMS. VisualStudioCode (VS Code) – це сучасне середовище розробки, яке забезпечує повний контроль над процесом створення веб-блогу. Використання VS Code у розробці веб-блогу має такі переваги:

Гнучкість у виборі технологій VS Code дозволяє використовувати HTML, CSS, JavaScript, а також популярні фреймворки, такі як React, Vue.js або Angular для створення інтерактивних елементів.

Підтримка Git та розширень, що дозволяє легко керувати версіями коду та розгорнути проєкт на різних хостингах.

Інтерактивність та кастомізація надає можливість програмістам розробляти унікальні рішення, що повністю відповідають вимогам проєкту.

Проста інтеграція з API та базами даних MySQL, MongoDB та Firebase дозволяє зберігати інформацію подій роботи блогу.

Порівняльна таблиця особливостей середовищ розробки веб-проєктів дозволяє наочно оцінити ключові відмінності між WordPress і VisualStudioCode. З точки зору зручності, гнучкості та вимог до технічної підготовки (таблиця 1.1). Як показано в таблиці, WordPress забезпечує швидкий старт завдяки готовим рішенням, тоді як VisualStudioCode надає повний контроль над кодом і дизайном.

Таблиця 1.1 Порівняльна таблиця особливостей середовищ розробки веб-проєктів

Платформа

Гнучкість

Простота у використанні

Можливість кастомізації

Підтримка інтерактивних елементів

WordPress

Середня

Висока

Низька без плагінів

Середня

Wix

Низька

Висока

Низька

Низька

Joomla

Висока

Середня

Висока

Середня

Ghost

Середня

Висока

Низька

Низька

VisualStudioCode

Висока

Низька (потрібні знання коду)

Висока

Висока

Висновки щодо вибору середовища розробки

У процесі вибору середовища розробки для створення інтерактивного веб-блогу було розглянуто кілька популярних платформ і інструментів. Основними критеріями для аналізу стали зручність роботи, підтримка сучасних веб-технологій, сумісність із необхідними бібліотеками, продуктивність та можливість розширення функціоналу.

На основі проведеного аналізу було обрано VisualStudioCode (VS Code) як основне середовище розробки. Основними перевагами цього інструменту є:

Кросплатформеність – підтримка Windows, Linux та macOS.

Гнучкість і розширюваність – велика кількість плагінів для **39** роботи з HTML, CSS, JavaScript, PHP, React та іншими технологіями.

Зручність роботи з Git – інтеграція системи контролю версій для командної роботи та збереження змін.

Інтеграція з серверами – можливість швидкого запуску локального сервера для тестування веб-застосунку.

Легка настройка – підтримка конфігураційного файлу settings.json, що дозволяє адаптувати середовище під індивідуальні потреби розробника.

**42** Таким чином на основі розглянутого матеріалу нами було вибране середовище розробки VisualStudioCode у зв'язку з тим, що на теперішній час дана мультиплатформа організована з максимальною апаратною і програмною сумісністю багатьох структурних компонентів інформаційної інфраструктури і таким чином дозволяє не тільки зручно і легко розробляти програмні проекти, а й надає можливість гармонійного інтегрування даних проектів для розвитку цієї ж інформаційної сервісної інфраструктури.

## 2 ПРОЄКТУВАННЯ ІНТЕРАКТИВНОГО ВЕБ-БЛОГУ

### 2.1 Вимоги до функціональності, дизайну та архітектури

Проєктування інтерактивного веб-блогу для реклами цифрової техніки передбачає визначення чітких вимог до його функціональності, дизайну та архітектури.

Встановлення цих вимог забезпечує ефективне функціонування веб-застосунку, високу продуктивність, безпеку та зручність для користувачів.

#### 2.1.1 Функціональні вимоги

Функціональні вимоги визначають основний набір можливостей веб-блогу, що забезпечуватимуть його ефективну роботу. Одним з основних функціональних компонентів веб-застосунку є система управління користувачами.

Система управління користувачами забезпечує зручну та безпечну взаємодію користувачів із веб-блогом. Реєстрація та автентифікація відбуваються за допомогою електронної пошти або акаунтів соціальних мереж, що спрощує доступ до платформи. Для безпеки використовується авторизація на основі JSON WebToken (JWT), що захищає

персональні дані користувачів. У разі втрати пароля передбачено функцію його відновлення через електронну пошту. Адміністратор має можливість розмежовувати права доступу, надаючи різні рівні привілеїв для звичайних користувачів та модераторів.

Система контролює автентичність користувачів, дозволяючи лише авторизованим особам виконувати певні дії, такі як коментування або створення контенту. Для запобігання несанкціонованого доступу застосовуються сучасні методи шифрування даних. Адміністратори можуть переглядати список користувачів, блокувати порушників правил та редагувати профілі. Усі дії користувачів логуються, що дає змогу аналізувати активність і виявляти можливі загрози безпеці. Гнучкість та масштабованість цієї системи дозволяє легко адаптувати її до потреб проєкту та зростання кількості користувачів.

Система управління контентом у веб-блозі дозволяє адміністраторам додавати, редагувати та видаляти публікації. Для зручного керування матеріалами передбачено можливість завантаження зображень, відео та інших медіафайлів. Статті можна розподіляти за категоріями, що полегшує пошук інформації для користувачів. Кожна публікація автоматично містить дані про автора та дату створення. Адміністратори можуть змінювати вміст статей без потреби залучення розробників. Контент організований у структурованому форматі, що забезпечує його зручне відображення на сайті. Всі ці функції сприяють ефективному управлінню блогом та покращують взаємодію з відвідувачами.

Система комунікації у веб-блозі дозволяє користувачам залишати коментарі під публікаціями, що сприяє активному обговоренню контенту. Реалізована система оцінювання у вигляді лайків і дизлайків допомагає визначити популярність статей та реакцію аудиторії. Функціонал відповіді на коментарі дає змогу вести дискусії та створювати гілки обговорень. Для підтримки якісного контенту адміністраторам надається можливість модерації коментарів. Це допомагає запобігти спаму, образам і неналежному контенту. Завдяки такій комунікаційній системі веб-блог стає більш інтерактивним і залучає користувачів до активної участі. Взаємодія між читачами формує спільноту навколо блогу та підвищує рівень залученості аудиторії.

Система навігації у веб-блозі забезпечує зручний пошук інформації за ключовими словами та категоріями. Функціонал фільтрації публікацій дозволяє сортувати контент за датою, популярністю та іншими критеріями. Інтуїтивно зрозуміле меню містить основні розділи сайту, що спрощує переміщення між сторінками. Завдяки добре структурованій навігації користувачі швидко знаходять потрібні статті та матеріали. Ефективний пошук і категоризація контенту підвищують зручність користування платформою. Система фільтрів дає змогу персоналізувати відображення публікацій відповідно до вподобань користувача. Це 29 сприяє покращенню користувацького

досвіду та залученню більшої кількості відвідувачів.

SEO-оптимізація веб-блогу сприяє покращенню його видимості **7** у пошукових системах. Автоматична генерація мета-даних, таких як title, description і keywords, допомагає підвищити рейтинг сторінок у результатах пошуку. Інтеграція з GoogleAnalytics дозволяє аналізувати поведінку **7** користувачів на сайті. Збір даних про відвідуваність, тривалість сесій і джерела трафіку **7** допомагає оптимізувати контент і маркетингові стратегії. Моніторинг популярних сторінок дозволяє виявляти найбільш ефективні матеріали та вдосконалювати контент-план. Використання аналітичних інструментів допомагає підвищити залученість аудиторії та покращити користувацький досвід. Оптимізований веб-блог сприяє зростанню органічного трафіку та розширенню аудиторії.

Адаптивність та продуктивність веб-блогу є ключовими факторами для забезпечення комфортного користування. Оптимізація зображень і медіафайлів дозволяє зменшити час завантаження сторінок. Застосування сучасних форматів, таких як WebP, допомагає знизити вагу файлів без втрати якості. Використання кешування дозволяє скоротити кількість запитів до сервера та прискорити завантаження контенту. Ефективне управління ресурсами сприяє стабільній роботі веб-сайту навіть при високому навантаженні. Адаптивний дизайн **30** забезпечує коректне відображення контенту на різних пристроях, включаючи смартфони та планшети. Усі ці заходи сприяють покращенню користувацького досвіду та зниженню рівня відмов.

Інтеграція з соціальними мережами дозволяє користувачам швидко поширювати контент. Додавання кнопок для публікації у Facebook, Instagram та Twitter сприяє збільшенню охоплення аудиторії. Віджет останніх новин із соціальних платформ допомагає відображати актуальні оновлення. Автоматична публікація нових статей **7** у соціальних мережах підвищує залученість користувачів. Взаємодія з соцмережами покращує SEO-оптимізацію та впізнаваність бренду. Інтеграція коментарів із соцмереж спрощує авторизацію та зменшує анонімність. Такі функції створюють зручну екосистему для активного обговорення контенту.

### 2.1.2 Вимоги до дизайну

Веб-блог повинен мати сучасний, привабливий та адаптивний дизайн, що забезпечить зручність користувачів і відповідатиме сучасним стандартам веб-розробки. Основні вимоги до дизайну включають:

Мінімалістичний дизайн інтерфейсу користувача забезпечує зручність сприйняття контенту. Контрастні кольори для основних елементів управління допомагають покращити навігацію. Чітке структурування сторінок дозволяє користувачам швидко

знаходити необхідну інформацію. Поділ інтерфейсу на логічні блоки сприяє комфортному сприйняттю матеріалів. Використання зручних шрифтів підвищує читабельність як на мобільних, так і на десктопних пристроях. Оптимізований UI сприяє швидкій адаптації нових користувачів. Гармонійне поєднання дизайну та функціональності робить веб-блог привабливим і доступним.

Адаптивність веб-блогу забезпечує його коректне відображення на мобільних пристроях і планшетах. Оптимізація для різних розмірів екранів дозволяє зберегти зручність користування незалежно від пристрою. Гнучке компонування елементів сторінки сприяє зручному розташуванню контенту та елементів управління. Динамічне налаштування макету дає можливість автоматично підлаштовуватися під ширину екрану. Використання адаптивних шрифтів і зображень покращує візуальне сприйняття контенту. Забезпечення швидкого завантаження сторінок на будь-якому пристрої підвищує рівень користувацького досвіду. Сучасні технології адаптивного дизайну сприяють комфортному перегляду веб-блогу в будь-яких умовах.

Зручна навігація відіграє важливу роль у покращенні користувацького досвіду. Інтуїтивне меню забезпечує швидкий **40** доступ до основних розділів сайту, дозволяючи легко знаходити необхідний контент. Використання бокових панелей допомагає компактно розміщувати додаткові елементи управління. Випадаючі списки дозволяють ефективно організувати вміст без перевантаження інтерфейсу. Виділення активних сторінок візуальними маркерами допомагає користувачам зрозуміти, де вони знаходяться. Чітка структура навігації сприяє зменшенню часу на пошук потрібної інформації. Продуманий дизайн меню робить взаємодію з веб-блогом комфортною та ефективною.

Логічна організація навігаційної системи забезпечує швидке орієнтування вмістом сайту. Інтуїтивно зрозуміле меню зменшує когнітивне навантаження користувачів, дозволяючи їм швидко знаходити потрібні розділи. Бокові панелі створюють додаткові можливості для доступу до категорій або популярних матеріалів. Випадаючі списки допомагають компактно розміщувати підкатегорії та тематичні рубрики. Виділення активної сторінки дозволяє уникнути плутанини та покращує взаємодію користувачів із сайтом. Використання різних кольорових маркерів або підсвічування активних елементів підвищує зручність навігації. Загалом, добре продумана структура дозволяє користувачам витратити мінімум часу на пошук інформації та максимально зосереджуватись на контенті.

### 2.1.3 Вимоги до архітектури

Архітектура веб-блогу базується на клієнт-серверній моделі, що забезпечує ефективну взаємодію між користувачем та сервером. Основні компоненти архітектури:

Клієнтська частина проєкту розробляється з використанням React.js, що забезпечує створення динамічного та інтерактивного інтерфейсу. Для адаптивного дизайну, який коректно відображається на різних пристроях, застосовуються Tailwind CSS або Bootstrap. Оптимізація відображення контенту досягається через кешування сторінок, що покращує швидкість завантаження та продуктивність. Використання React.js дозволяє створювати компоненти інтерфейсу, які можна повторно використовувати, що спрощує розробку та підтримку. Tailwind CSS або Bootstrap надають готові стилі та компоненти, що прискорює процес розробки та забезпечує єдиний стиль. Кешування сторінок зберігає статичні ресурси на стороні клієнта, зменшуючи кількість запитів до сервера та час очікування. Загалом, ці технології та методи оптимізації спрямовані на створення швидкого, адаптивного та зручного для користувача веб-інтерфейсу.

Серверна частина проєкту розробляється з використанням Node.js та Express.js, що дозволяє ефективно обробляти запити від клієнта. Node.js забезпечує асинхронну обробку, що покращує продуктивність серверної частини. Express.js спрощує створення веб-додатків та API, надаючи необхідні інструменти та функції. Для взаємодії між клієнтом і сервером реалізується API, який дозволяє обмінюватися даними у структурованому форматі. Захист **33** від несанкціонованого доступу забезпечується за допомогою JWT (JSON WebTokens), що гарантує безпеку передачі даних. Використання JWT дозволяє перевіряти автентичність запитів та авторизувати користувачів. Загалом, ці технології та методи захисту спрямовані на створення надійної, безпечної та ефективної серверної частини.

Для зберігання даних проєкту використовується MongoDB або MySQL, вибір між якими залежить від конкретних вимог. MongoDB організовує дані у вигляді колекцій, тоді як MySQL використовує таблиці для структурування інформації. Обидві системи використовуються для збереження даних про користувачів, статей та коментарів. Для забезпечення швидкого пошуку даних застосовується індексація, що оптимізує запити до бази даних. MongoDB є NoSQL **18** базою даних, яка забезпечує гнучкість схеми та масштабованість, тоді як MySQL - реляційна база даних, що забезпечує цілісність даних. Вибір між MongoDB та MySQL залежить від таких факторів, як складність структури даних, вимоги до масштабованості та необхідність транзакційності. Індексація дозволяє значно прискорити операції пошуку, сортування та фільтрації даних, що **33** покращує загальну продуктивність системи.

**22** Для забезпечення безпеки даних користувачів використовується шифрування за допомогою алгоритму bcrypt, що робить паролі нечитабельними для зловмисників. Таким чином Веб-блог захищений від поширених веб-атак, таких як SQL-ін'єкції та XSS-атаки, що запобігає несанкціонованому доступу до бази даних та впровадженню шкідливого коду. Захист від SQL-ін'єкцій та XSS-атак запобігає викраденню даних та пошкодженню веб-сайту. Шифрування bcrypt забезпечує надійний захист паролів,

навіть якщо база даних буде зламана. Використання протоколу HTTPS гарантує безпечний обмін **18** даними між клієнтом та сервером, шифруючи всю передану інформацію. HTTPS забезпечує конфіденційність даних користувачів, запобігаючи перехопленню інформації під час передачі. Загалом, ці заходи безпеки спрямовані на створення надійного та безпечного веб-блогу, який захищає дані користувачів та забезпечує безпечний обмін інформацією.

Клієнтська частина веб-блогу розгортається на платформах Vercel або Netlify, що забезпечує швидке та ефективно розміщення статичних файлів. Серверна частина веб-блогу розміщується на Heroku або AWS, що дозволяє масштабувати додаток залежно від навантаження. Автоматичне оновлення веб-блогу здійснюється за допомогою GitHubActions, що **32** спрощує процес розгортання та зменшує ризик помилок. Використання Vercel або Netlify **22** дозволяє автоматично розгортати зміни з репозиторію GitHub, що прискорює процес розробки. Хостинг серверної частини на Heroku або AWS забезпечує **5** надійність та стабільність роботи веб-блогу. GitHubActions дозволяє автоматизувати тестування та розгортання, що зменшує час на ручну роботу та покращує якість коду. Загалом, ці технології та методи розгортання та підтримки спрямовані на створення надійного, масштабованого та легко підтримуваного веб-блогу.

Визначення функціональних, дизайнерських та архітектурних вимог є важливим етапом розробки інтерактивного веб-блогу. Врахування цих вимог забезпечує ефективність веб-застосунку, зручність користувачів, швидкість завантаження сторінок та безпеку обробки даних. Використання сучасних веб-технологій сприятиме створенню продуктивного та привабливого онлайн-ресурсу, що відповідатиме вимогам користувачів і бізнес-цілям проєкту.

## 2.2 Вибір технологій та інструментів (HTML, CSS, JavaScript, фреймворки)

Розробка інтерактивного веб-блогу для реклами цифрової техніки потребує **5** використання сучасних технологій та інструментів, які забезпечать ефективність, продуктивність і зручність роботи веб-застосунку. **5** Для реалізації проєкту були обрані такі технології: HTML, CSS, JavaScript, а також відповідні фреймворки та бібліотеки.

HTML використовується **5** для створення структури сторінок, CSS — для оформлення зовнішнього вигляду та адаптивності, а JavaScript — для забезпечення динамічної взаємодії з користувачем. Для підвищення продуктивності та спрощення розробки застосовуються популярні фреймворки та бібліотеки, такі як Bootstrap для стилізації інтерфейсу, а також jQuery або сучасні JavaScript-фреймворки для реалізації інтерактивних елементів. Це дозволяє створити зручний, функціональний і сучасний веб-блог, орієнтований на ефективну презентацію цифрової техніки.

## 2.2.1 Технології для фронтенду

Фронтенд, або клієнтська частина, відповідає за візуальне відображення інтерфейсу веб-блогу, роблячи його привабливим та зручним для користувачів. Він забезпечує інтерактивність, дозволяючи користувачам взаємодіяти з елементами сайту, такими як коментарі, лайки та кнопки навігації. Фронтенд обробляє введення користувача, наприклад, введення тексту в поле пошуку або заповнення форми для коментарів. Він відповідає за динамічне оновлення контенту на сторінці без перезавантаження, покращуючи користувацький досвід. Фронтенд забезпечує адаптивність веб-блогу, щоб він коректно відображався **11** на різних пристроях, від комп'ютерів до мобільних телефонів. Він використовує **11** технології, такі як HTML, CSS та JavaScript, для створення структури, стилів та інтерактивності веб-блогу. Загалом, фронтенд **8** відіграє ключову роль у створенні зручного та привабливого інтерфейсу, який забезпечує ефективну взаємодію користувачів з веб-блогом.

HTML, як основа будь-якої веб-сторінки, визначає структуру веб-блогу, забезпечуючи його правильне відображення в браузері. Використання семантичних тегів, таких як <header>, <main>, <section>, <article>, <aside> та <footer>, покращує доступність веб-блогу **15** для людей з обмеженими можливостями та сприяє кращій індексації пошуковими системами. HTML дозволяє формувати чітку структуру контенту веб-блогу, використовуючи заголовки, списки, таблиці та форми для зручного представлення інформації. **8** За допомогою HTML можна вставляти мультимедійні елементи, такі як зображення, відео та аудіо, що робить контент веб-блогу більш привабливим та інформативним. Семантичні теги **11** допомагають браузерам та пошуковим системам краще розуміти зміст веб-блогу, що покращує його SEO-показники. HTML забезпечує можливість створення інтерактивних елементів, таких як форми для коментарів та пошуку, що покращує взаємодію користувачів з веб-блогом. Загалом, HTML є невід'ємною частиною розробки веб-блогу, забезпечуючи його структуру, доступність та функціональність.

CSS, або CascadingStyleSheets, використовується для оформлення веб-блогу, забезпечуючи стильове оформлення та адаптивність інтерфейсу. **16** CSS Grid і Flexbox дозволяють створювати гнучкі та складні макети сторінок, що покращує розташування елементів на веб-блосі. Завдяки Mediaqueries, веб-блог **15** адаптується до різних розмірів екранів, забезпечуючи коректне відображення **16** на мобільних пристроях. Використання Mediaqueries дозволяє створювати різні стилі **19** для різних пристроїв, що покращує користувацький досвід. Анімації та переходи, **8** створені за допомогою CSS transitions та keyframes, додають плавні візуальні ефекти, роблячи веб-блог більш привабливим. CSS дозволяє контролювати кольори, шрифти, відступи та інші стильові елементи, що забезпечує єдиний стиль веб-блогу. Загалом, **19** CSS відіграє важливу роль у створенні привабливого, адаптивного та зручного для користувача веб-блогу.

Для прискорення розробки використовується Tailwind CSS або Bootstrap – популярні CSS-фреймворки, що містять готові стилізовані компоненти.

JavaScript є ключовою мовою програмування для додавання інтерактивності до веб-блогу, забезпечуючи **17** динамічну взаємодію з користувачем. **28** За допомогою JavaScript можна обробляти події, такі як натискання кнопок та відправлення форм, що дозволяє **17** реагувати на дії користувачів у реальному часі. JavaScript **28** дозволяє динамічно оновлювати вміст веб-блогу **13** без перезавантаження сторінки, що покращує користувацький досвід та зменшує час очікування. Використання AJAX дозволяє JavaScript асинхронно обмінюватися даними з сервером, що дозволяє завантажувати новий контент без переривання взаємодії користувача з веб-блогом. JavaScript дозволяє створювати складні інтерактивні елементи, такі як фільтри пошуку, слайдери та анімації, що робить веб-блог більш привабливим та функціональним. JavaScript дозволяє реалізовувати валідацію форм **13** на клієнтській стороні, що зменшує навантаження на сервер та покращує користувацький досвід. Загалом, **20** JavaScript відіграє важливу роль у створенні інтерактивного та динамічного веб-блогу, забезпечуючи ефективну взаємодію користувачів з сайтом.

React.js **20** використовується для створення динамічного інтерфейсу веб-блогу, забезпечуючи високу продуктивність та інтерактивність. Завдяки віртуальному DOM, React.js мінімізує кількість операцій з реальним DOM, що значно покращує продуктивність веб-блогу. Компонентний підхід React.js дозволяє створювати повторно використовувані елементи інтерфейсу, що спрощує розробку та підтримку веб-блогу. Хуки, такі як useState та useEffect, роблять розробку компонентів React.js більш зручною та ефективною, дозволяючи легко керувати станом та побічними ефектами. Для управління станом веб-блогу використовуються Redux або Context API, що дозволяє ефективно обмінюватися даними між компонентами. React.js **6** дозволяє створювати складні інтерфейси з мінімальною кількістю коду, що робить розробку веб-блогу швидкою та ефективною. **6** Загалом, React.js є потужним інструментом для створення динамічного **1** та інтерактивного веб-блогу, забезпечуючи високу продуктивність та зручність розробки. Для управління станом застосунку може використовуватись.

Redux або Context API використовуються для ефективного управління станом веб-блогу, дозволяючи передавати дані між компонентами без зайвого дублювання коду. Ці інструменти забезпечують централізоване сховище стану, що полегшує доступ до даних з будь-якого компонента веб-блогу. Redux, як більш потужний інструмент, підходить для складних веб-блогів з великою кількістю даних та взаємодій між компонентами. Context API, як простіший інструмент, добре підходить для невеликих та середніх веб-блогів, де не потрібні складні механізми управління станом. Використання Redux або Context API дозволяє уникнути передачі даних через props, що спрощує структуру компонентів та покращує читабельність коду. Ці інструменти дозволяють

легко відстежувати зміни стану веб-блогу, що спрощує налагодження та підтримку коду. Загалом, Redux або Context API є важливими інструментами для створення ефективного та масштабованого веб-блогу, забезпечуючи зручне **9** управління станом та обмін даними між компонентами.

### 2.2.2 Технології для бекенду

Бекенд (серверна частина) обробляє запити користувачів, зберігає та обробляє дані, забезпечує безпеку та взаємодію **27** з базою даних. Node.js, як середовище виконання JavaScript на **9** сервері, забезпечує швидку та **10** ефективну обробку запитів від клієнта. **9** Express.js, фреймворк для Node.js, спрощує розробку серверної частини, надаючи інструменти для обробки HTTP-запитів (GET, POST, PUT, DELETE). Маршрутизація, яку забезпечує Express.js, дозволяє створювати RESTful API для обміну даними **9** між клієнтом та сервером. Node.js та Express.js дозволяють легко підключатися до бази даних та управляти користувачами веб-блогу, забезпечуючи надійне зберігання та обробку даних. Використання Node.js **10** дозволяє розробляти серверні додатки на JavaScript, що спрощує розробку для фронтенд-розробників. Express.js надає middleware, **14** що дозволяє додавати додаткову функціональність до серверної частини, таку як аутентифікація та авторизація. Загалом, Node.js та Express.js є потужними інструментами для створення швидкої, надійної та масштабованої серверної частини веб-блогу.

**1** Для зберігання даних веб-блогу, таких як статті, користувачі та коментарі, використовується база даних, вибір якої залежить від потреб проекту. MySQL обирається, якщо потрібна реляційна структура даних, де дані зберігаються у вигляді таблиць зі зв'язками між ними. MongoDB використовується, якщо необхідна гнучка документна структура, де **14** дані зберігаються у форматі JSON, що дозволяє легко зберігати неструктурований контент. MySQL забезпечує надійність та **26** цілісність даних завдяки підтримці транзакцій та обмежень цілісності. MySQL добре підходить для веб-блогу з чітко структурованими даними, такими як користувачі та коментарі, де потрібні складні запити та зв'язки між таблицями. MongoDB **10** забезпечує гнучкість та масштабованість, дозволяючи легко додавати нові поля та змінювати структуру даних без зміни схеми. MongoDB добре підходить для веб-блогу **26** з великою кількістю контенту, такого як статті, де потрібна гнучкість та швидкість обробки даних у форматі JSON.

### Інструменти для розробки

VisualStudioCode (VS Code) є основним середовищем розробки, що забезпечує зручне написання та редагування коду. VS Code підтримує розширення для React, Node.js та Git, **1** що робить його універсальним інструментом для розробки веб-блогу. VS Code дозволяє використовувати інтегрований термінал, що спрощує виконання команд Git

та інших інструментів. Розширення для VS Code, такі як ESLint та Prettier, допомагають підтримувати єдиний стиль коду та запобігають помилкам. VS Code має вбудований налагоджувач для JavaScript та Node.js, що спрощує пошук та виправлення помилок.

**23** Git та GitHub використовуються для контролю версій коду, що дозволяє відстежувати зміни та повертатися до попередніх версій. GitHub забезпечує платформу для спільної роботи над проектом, **44** дозволяючи кільком розробникам працювати над одним кодом одночасно. GitHub дозволяє створювати пул-реквести для перевірки коду перед його злиттям з основною гілкою. Git дозволяє створювати гілки для розробки нових **23** функцій, що **43** дозволяє працювати над кількома завданнями одночасно. Також Git дозволяє відстежувати зміни в коді з коментарями, що полегшує розуміння історії розробки.

Postman використовується для тестування API-запитів, що дозволяє перевірити працездатність серверної частини веб-блогу. Postman дозволяє створювати колекції запитів для тестування різних сценаріїв використання API.

Docker використовується для контейнеризації застосунку, що дозволяє створити ізольоване середовище для розгортання веб-блогу на сервері. Docker дозволяє легко розгортати веб-блог на різних серверах, забезпечуючи однаковість середовища. Docker дозволяє створювати образи для різних компонентів веб-блогу, що спрощує масштабування та управління застосунком

Розгортання проекту

Vercel або Netlify використовуються для швидкого та ефективного розгортання клієнтської частини веб-блогу, написаної на React.js. Ці сервіси забезпечують автоматичне розгортання при кожному оновленні коду в репозиторії Git, що спрощує процес публікації. Vercel та Netlify забезпечують CDN (ContentDeliveryNetwork) для швидкої доставки статичних файлів веб-блогу користувачам з усього світу.

Heroku або AWS використовуються для хостингу серверної частини веб-блогу, написаної на Node.js та Express.js, забезпечуючи надійну роботу серверу. AWS надає широкий спектр сервісів для масштабування та управління сервером, **38** що дозволяє обробляти велику кількість запитів. Heroku пропонує просту у використанні платформу для розгортання Node.js додатків, що спрощує процес публікації серверної частини. Heroku та AWS надають інструменти для моніторингу та логування серверної частини, що **37** дозволяє швидко виявляти та виправляти помилки.

**37** Cloudinary використовується **1** для зберігання та оптимізації зображень та мультимедійних файлів, що покращує швидкість завантаження веб-блогу. Цей сервіс дозволяє автоматично змінювати розмір та формат зображень, що адаптує їх для

різних пристроїв. Cloudinary дозволяє зберігати відео та аудіо файли, що розширює можливості веб-блогу для публікації різноманітного контенту.

Використання цих сервісів дозволяє зосередитися на розробці веб-блогу, а не на управлінні серверами та інфраструктурою. Загалом, ці сервіси забезпечують розробнику надійне, швидке та масштабоване розгортання веб-блогу в Інтернеті.

Обрані технології забезпечують високу продуктивність, гнучкість у розробці та легкість у підтримці веб-блогу.

Розробка макету та інтерактивних елементів UX/UI

Розробка веб-блогу для реклами цифрової техніки передбачає створення зручного та естетично привабливого інтерфейсу користувача. UX/UI-дизайн **1** відіграє ключову роль у забезпеченні позитивного досвіду взаємодії користувачів із платформою.

У цьому розділі розглядаються принципи розробки макету веб-блогу, підхід до UI-дизайну та інтерактивних елементів, які підвищують зручність користування.

### 2.3.1 UX-дизайн: зручність та навігація

UX, або UserExperience, визначає комфорт та зрозумілість інтерфейсу веб-блогу для користувача, що є ключовим фактором його успіху. UX-дизайн веб-блогу повинен враховувати потреби та очікування цільової аудиторії, щоб забезпечити максимальний комфорт користування. Принцип простоти вимагає мінімалістичного дизайну без зайвих елементів, що відволікають користувача від основного контенту веб-блогу. Логічна структура забезпечує зручну навігацію, що дозволяє користувачам швидко знаходити потрібну інформацію на веб-близі. Адаптивність гарантує правильне відображення веб-блогу **1** на різних пристроях, включаючи мобільні телефони, планшети та комп'ютери. Інтерактивність **1** покращує користувацький досвід за допомогою динамічних елементів, таких як анімації та переходи. Адаптивний дизайн забезпечує зручний доступ до веб-блогу з будь-якого пристрою, що збільшує аудиторію. Продумана структура веб-блогу дозволяє користувачам легко знаходити статті, коментарі та іншу інформацію. Використання інтерактивних елементів робить веб-блог більш привабливим та залучає користувачів до активної взаємодії. Загалом, UX-дизайн відіграє важливу роль у створенні зручного, зрозумілого та привабливого веб-блогу, який забезпечує позитивний досвід для користувачів.

### 2.3.2 UI-дизайн

UI (UserInterface) – це візуальна складова, яка формує зовнішній вигляд веб-блогу. При створенні макету застосовувалися професійні інструменти UI-дизайну, такі як Figma та Adobe XD, **6** що дозволяє створювати прототипи та перевіряти дизайн перед

реалізацією у кодї.

Головна сторінка веб-блогу починається з логотипу та заголовка сайту, які визначають його ідентичність та привертають увагу користувачів. Головне меню з категоріями, такими як "Новинки", "Огляди", "Порівняння" та "Відгуки", **36** забезпечує зручну навігацію та швидкий доступ до потрібної інформації. Головний банер з актуальними акціями та новинами розміщується у верхній частині сторінки, щоб привернути увагу користувачів до важливих подій. Головний банер повинен бути яскравим та інформативним, щоб привернути увагу користувачів та спонукати їх до дій. Логотип та заголовок сайту повинні бути чіткими та лаконічними, щоб користувачі могли швидко зрозуміти, на який сайт вони потрапили. Головне меню повинно бути структурованим та зрозумілим, щоб користувачі могли легко знайти потрібну категорію. Загалом, головна сторінка веб-блогу повинна бути привабливою, зручною та інформативною, щоб забезпечити позитивний користувацький досвід.

Сторінка статті починається з великого заголовка, який привертає увагу користувачів та відображає основну тему статті. Дата публікації та автор статті відображаються під заголовком, щоб надати користувачам контекст та інформацію про походження статті. Великий заголовок повинен бути чітким та інформативним, щоб користувачі могли швидко зрозуміти, про що йдеться в статті. Основний контент статті, що складається з тексту, зображень та відео, займає центральну частину сторінки, забезпечуючи зручне читання та перегляд. Основний контент статті повинен бути структурованим та легко читабельним, щоб користувачі могли швидко отримати потрібну інформацію. Блок коментарів розташований внизу сторінки, щоб користувачі могли ділитися своїми думками та взаємодіяти з автором та іншими читачами. Блок коментарів повинен бути зручним та функціональним, щоб користувачі могли легко залишати коментарі та брати участь у дискусіях.

Панель навігації забезпечує зручний доступ до основних розділів веб-блогу, покращуючи користувацький досвід. Меню категорій **3** дозволяє користувачам швидко переходити між розділами, такими як "Новинки", "Огляди" та "Відгуки". Меню категорій повинно бути чітким та структурованим, щоб користувачі могли легко орієнтуватися на сайті. Пошук по сайту з інтерактивним автозаповненням запитів допомагає **3** користувачам швидко знаходити потрібну інформацію. Пошук по сайту повинен бути швидким та точним, щоб користувачі могли ефективно знаходити потрібну інформацію. Інтерактивне автозаповнення запитів допомагає користувачам швидше вводити пошукові запити та отримувати релевантні результати. Загалом, панель навігації повинна бути зручною, інтуїтивно зрозумілою та ефективною, щоб забезпечити позитивний користувацький досвід.

Форма зворотного зв'язку містить поля введення для імені, email та тексту

повідомлення, що **3** дозволяє користувачам легко зв'язатися **3** з адміністрацією веб-блогу. Поля введення **3** повинні бути чіткими та зрозумілими, щоб користувачі могли легко заповнити форму. Кнопка «Надіслати» взаємодіє із сервером через AJAX, що дозволяє відправляти повідомлення **1** без перезавантаження сторінки. Кнопка «Надіслати» повинна бути помітною та легко натискатися, щоб користувачі могли швидко відправити повідомлення. Візуальний стиль веб-блогу повинен бути сучасним та лаконічним, щоб забезпечити приємний та зручний досвід користування. Візуальний стиль веб-блогу повинен відповідати тематиці та цільовій аудиторії, щоб створити привабливий та відповідний інтерфейс. Загалом, форма зворотного зв'язку повинна бути зручною та функціональною, а візуальний стиль – привабливим та відповідним, щоб забезпечити позитивний користувацький досвід.

Основна кольорова гама веб-блогу базується на поєднанні темних і світлих тонів для забезпечення контрастності та зручного сприйняття контенту. Контрастність між темними та світлими тонами дозволяє виділити важливі елементи веб-блогу та полегшити читання тексту. Вибір читабельних шрифтів забезпечує комфортне читання контенту, навіть на невеликих екранах мобільних пристроїв. Використання читабельних шрифтів, таких як Roboto або OpenSans, покращує досвід користувачів, які читають довгі статті або коментарі. SVG-іконки використовуються для навігації та інтерактивних елементів, забезпечуючи чітке та масштабоване відображення **1** на різних пристроях. Використання SVG-іконок дозволяє створювати сучасний та стильний дизайн веб-блогу, забезпечуючи чітке відображення елементів інтерфейсу.

Інтерактивні елементи та їх реалізація

Hover-ефекти для кнопок і посилань, такі як зміна кольору або збільшення, покращують взаємодію користувачів з інтерфейсом веб-блогу. Hover-ефекти роблять інтерфейс веб-блогу більш динамічним та візуально привабливим.

Lazy-loading для зображень забезпечує відкладене завантаження, що зменшує час завантаження сторінки та покращує продуктивність веб-блогу. Lazy-loading зменшує навантаження на сервер та прискорює завантаження сторінки, особливо на мобільних пристроях.

Слайдери та каруселі використовуються для відображення популярних товарів або оглядів, що робить веб-блог більш привабливим та інформативним. Слайдери та каруселі дозволяють ефективно демонструвати велику кількість контенту на обмеженому просторі.

Модальні вікна дозволяють швидко переглядати інформацію без переходу на нову сторінку, що **1** покращує користувацький досвід. Модальні вікна дозволяють швидко

отримувати додаткову інформацію без переривання процесу перегляду веб-блогу.

Фільтрація та сортування статей за популярністю або датою дозволяє користувачам легко знаходити потрібну інформацію. Фільтрація та сортування статей дозволяє користувачам швидко знаходити релевантний контент.

Система лайків і коментарів з асинхронним оновленням через API дозволяє користувачам активно взаємодіяти з контентом веб-блогу. Асинхронне оновлення лайків і коментарів дозволяє користувачам бачити зміни в реальному часі **1** без перезавантаження сторінки.

**1** Використання JavaScript **6** дозволяє створювати складні інтерактивні елементи, які покращують користувацький досвід.

API дозволяє ефективно обмінюватися даними між клієнтом та сервером, забезпечуючи швидке оновлення контенту.

Для забезпечення коректного відображення веб-блогу на всіх пристроях використовуються підходи, що забезпечують адаптивність та кросбраузерність. Mediaqueries у CSS дозволяють адаптувати веб-блог **31** під різні розміри екранів, забезпечуючи зручне відображення на мобільних телефонах, планшетах та комп'ютерах. Тестування в різних браузерах, таких як Chrome, Firefox, Edge та Safari, дозволяє виявити та виправити можливі проблеми з відображенням веб-блогу, виявити та виправити проблеми з сумісністю, що забезпечує стабільну роботу веб-блогу. Також тестування в різних браузерах дозволяє забезпечити однакову якість відображення веб-блогу для всіх користувачів, незалежно від їхнього браузера. Перевірка швидкодії через GoogleLighthouse дозволяє оптимізувати веб-блог для швидкого завантаження **1** на різних пристроях та в різних браузерах. GoogleLighthouse надає рекомендації щодо оптимізації швидкодії веб-блогу, що покращує його SEO. Використання адаптивного дизайну дозволяє веб-блогу автоматично підлаштовуватися під розмір екрана пристрою, що забезпечує зручне читання та навігацію.

Отже продуманий UX/UI-дизайн **1** відіграє ключову роль у забезпеченні зручності та ефективності веб-блогу. Використання адаптивних технологій сприяє коректному відображенню ресурсу на різних пристроях, а застосування інтерактивних елементів покращує взаємодію користувачів із платформою.

## 3 РОЗРОБКА ІНТЕРАКТИВНОГО ВЕБ-БЛОГУ

### 3.1 Розробка веб-блогу в середовищі VisualStudioCode

Робота над блогом почалась з підготовки. Спочатку було визначено мету. Метою є

створення сайту, який дає змогу дізнатись інформацію про цифрову техніку. Також користувачі можуть порівняти ціни в різних магазинах та перейти за посиланням на сайти з товарами.

Далі необхідно було визначити аудиторію користування сайтом. Основна група – це молоді люди віком 18 – 35 років. Їх цікавить цифрова техніка. Також сайт стане у пригоді людям середнього віку, які хочуть купити техніку для дому чи роботи. Схема структури сайту наведена в таблиці 3.1.

Таблиця 3.1– Схема структури сайту.

Вікова група

Частка користувачів

Основні інтереси

Ймовірність користування сайтом

18–24 роки

35%

Новинки техніки, покупки онлайн

Висока

25–34 роки

30%

Порівняння цін, вибір техніки

Висока

35–44 роки

20%

Надійна техніка, покупки для сім'ї

Середня

45+ років

15%

Прості пристрої, консультації

Низька

Для стартової розробки проекту нами було обрано середовище VisualStudioCode (VS Code). Цей варіант дозволяє повністю контролювати структуру сайту, гнучко налаштовувати функціонал і реалізовувати дизайн згідно з власними потребами. Розробка веб-блогу спочатку в середовищі VisualStudioCode дала змогу глибше зрозуміти структуру веб-сторінки, верстку, роботу CSS-фреймворків і JavaScript. Це важливо для розуміння того, як побудований сайт «з нуля».

У процесі розробки було створено структуру директорій, що містить такі основні папки: index.html, styles/ для CSS-файлів, scripts/ для JavaScript, images/ для зображень. HTML-розмітку сторінки було створено вручну з використанням семантичних тегів: <header>, <main>, <section>, <article>, <footer>.

Сайт має просту структуру. На головній сторінці є коротка інформація про проєкт. Далі – сторінки з групами товарів: ноутбуки, принтери, системні блоки. Також є сторінка з посиланнями на магазини. Є розділ «Форум», де користувачі можуть писати свої думки та обговорювати техніку.

Структура сайту:

Головна сторінка (з описом сайту та гаслом);

Категорії товарів;

Підкатегорії з товарами (фото, опис, ціна);

Посилання на магазини;

Форум для обговорень.

Усі сторінки сайту об'єднані зручною навігацією, яка дозволяє швидко переходити між розділами. Було підготовлено візуальні матеріали. Зображення товарів, логотип, піктограми та кнопки створено в Canva, Figma та Photoshop. Для кольорів обрано світлі пастельні тони. Вони роблять сайт приємним на вигляд.

Для навігації створено меню: «Про сайт», «Ноутбуки», «Принтери», «Системні блоки», «Форум». З меню можна повернутись на головну сторінку. На рис. 3.1 наведено фрагмент HTML-коду головної сторінки

Рисунок 3.1 – Вигляд коду головної сторінки веб-блогу в середовищі

## VisualStudioCode

Весь текст перекладено українською. Це стосується заголовків, описів, назв кнопок, підказок. Такий підхід робить сайт зручнішим для користувачів.

Кожен розділ сторінки було логічно структуровано для кращого сприйняття інформації користувачем. Було застосовано адаптивний підхід до верстки, що дозволяє елементам автоматично підлаштовуватися під розмір екрана. Основна увага приділялась зручності навігації та візуальній привабливості інтерфейсу. Завдяки цьому сайт виглядає сучасно та однаково добре відображається як на десктопах, так і на мобільних пристроях.

Для стилізації застосовано фреймворк Tailwind CSS, що дозволяє швидко створювати адаптивний та сучасний інтерфейс. Усі стилі були створені з урахуванням адаптивності до різних типів пристроїв. На рис.3.2 наведено фрагмент HTML-коду картки товару.

Рисунок 3.2 – Фрагмент HTML-коду картки товару в середовищі

## VisualStudioCode

Для стилізації був застосований фреймворк Tailwind CSS, що дозволяє швидко створювати адаптивний та сучасний інтерфейс. Усі стилі були створені з урахуванням адаптивності до різних типів пристроїв. Tailwind CSS надає готові класи, які значно пришвидшують розробку без необхідності написання окремого CSS-коду. Завдяки утилітам типу flex, grid, shadow, rounded, text-center дизайн сайту виглядає сучасно та зрозуміло. Для мобільних пристроїв використовувалися класи з префіксами sm:, md:, lg:, що дозволяє налаштовувати вигляд компонентів у залежності від ширини екрана. Це забезпечує коректне відображення блогу на смартфонах, планшетах і десктопах. Tailwind CSS також сприяє підтримці єдиного стилю по всьому сайту завдяки системі кольорів, відступів і розмірів. Крім того, його інтеграція з VisualStudioCode через плагіни забезпечила підсвічування класів і автозаповнення, що підвищило швидкість і точність верстки.

JavaScript використано для реалізації інтерактивних елементів: слайдерів, обробки форм, переходів між сторінками без перезавантаження. На рис.3.3 наведений функціонал відображення дати публікації.

Рисунок 3.3 – Функціонал відображення дати публікації

Розробка веб-блогу в середовищі VisualStudioCode розпочалась із створення базової структури HTML-документа та підключення стилів. Особливу увагу було приділено зручності роботи з кодом та використанню розширень, які спрощують верстку та налагодження. Середовище VisualStudioCode забезпечило комфортну та гнучку роботу з усіма файлами проєкту. Завдяки цьому вдалося ефективно організувати процес

розробки з нуля.

Розробка в середовищі VS Code дала змогу краще контролювати розміщення елементів, реалізувати унікальний дизайн і перевірити адаптивність вручну через панель розробника браузера. Під час розробки також використовувалися системи сітки та компоновання, що дозволило досягти стабільного відображення в різних браузерах. Особливо корисним виявилось використання Live Server, який дозволяє переглядати зміни в режимі реального часу без оновлення сторінки вручну. Для оцінки швидкодії та оптимізації сторінок було задіяно сервіс PageSpeedInsights.

### 3.2 Модернізація веб-блогу на платформі WordPress

Для подальшої модернізації нами було обрано платформу WordPress. Вона проста у використанні, має багато тем та розширень, а також дозволяє адаптувати функціонування сайту на мобільних пристроях. Він підтримує українську мову. Інші платформи, такі як Joomla чи Drupal, складніші й вимагають більше знань.

#### 3.2.1 Реалізація інтерфейсу

На цьому етапі відбулася реалізація візуального вигляду веб-сайту, що включає верстку елементів сторінок, їх графічне оформлення (стилізацію) та адаптацію до різних типів пристроїв: настільних комп'ютерів, планшетів і смартфонів.

У зв'язку з цим перехід до WordPress дозволив оцінити переваги системи управління контентом (CMS) та ефективність готових інструментів у реальних умовах. Використання WordPress значно спростило процес створення повнофункціонального веб-блогу завдяки широкому вибору тем, плагінів і візуальних редакторів. Інтерфейс адміністративної панелі виявився інтуїтивно зрозумілим, що дало змогу швидко наповнювати сайт контентом, змінювати структуру меню, додавати категорії та керувати публікаціями без втручання в код. Завдяки цьому навіть користувачі без досвіду програмування можуть ефективно підтримувати й оновлювати сайт.

Окрім цього на базі WordPress було здійснено масштабування проекту – додано нові функції: реєстрація користувачів, додавання постів, коментарі, SEO, адаптивні теми тощо. Такий підхід дозволяє зосередитися не лише на технічній реалізації, а й на зручності для кінцевого користувача та візуальній привабливості блогу.

Хоча WordPress генерує HTML-код автоматично за допомогою блоків і шаблонів, для точного налаштування інтерфейсу було виконано ручну правку HTML-структури за допомогою вбудованого редактора блоків (Gutenberg) та використання кастомних HTML-блоків. Було обрано шаблон, що підтримує сучасну модульну верстку.

Для локального розгортання сайту на платформі WordPress, тестування його функціоналу, налагодження плагінів і тем, а також для імітації умов роботи вебсервера **2** без підключення до зовнішнього хостингу було задіяно локальне серверне середовище MAMP.

MAMP (Macintosh, Apache, MySQL, PHP) – це локальне серверне середовище, яке дозволяє запускати веб-сервер **2** на персональному комп'ютері без необхідності доступу до інтернету. Основна мета використання MAMP – створення, тестування та налагодження вебсайтів **2** у локальному середовищі до їх публікації в Інтернеті. MAMP включає всі необхідні компоненти для роботи з динамічними веб-сайтами: веб-сервер **2** Apache або Nginx, систему управління базами даних MySQL або MariaDB та інтерпретатор PHP. Завдяки цьому розробник отримує повноцінне серверне середовище для роботи з такими платформами, як WordPress, Drupal, Joomla тощо.

MAMP простий у встановленні та налаштуванні, підтримує роботу як на macOS, **2** так і на Windows, що робить його універсальним інструментом. Крім того, до складу MAMP входить веб-інтерфейс phpMyAdmin, який спрощує керування базами даних. MAMP дозволяє створювати кілька віртуальних хостів, що зручно для розробки кількох проєктів одночасно.

Хоча WordPress автоматично генерує HTML-код за допомогою шаблонів і блоків, для досягнення точного контролю над інтерфейсом було використано ручне редагування HTML-структури за допомогою вбудованого редактора блоків Gutenberg, а також створення кастомних HTML-блоків. Для проєкту було обрано шаблон, що підтримує сучасний підхід до модульної адаптивної верстки.

Основна структура сайту поділяється на такі ключові секції:

Шапка сайту (header): містить логотип проєкту, головне навігаційне меню з категоріями, а також пошуковий рядок для зручного доступу до контенту;

Основна частина (main): представлена блоками з товарами – кожен містить зображення, назву, короткий опис, ціну та інтерактивну кнопку для перегляду або покупки;

Верстка виконана з урахуванням правил семантики HTML5 – використано теги <section>, <article>, <nav>, <footer> тощо.

Для візуального оформлення інтерфейсу використовувалися як вбудовані інструменти теми WordPress, так і додаткові правила CSS. Загальний дизайн орієнтовано на мінімалізм, зручність користування та відповідність сучасним UI-трендам.

3.2.2 Основні принципи стилізації.

Для колірної гами обрано м'які пастельні відтінки (блакитний, світло-сірий, м'ятний), які створюють легкий, ненав'язливий інтерфейс, комфортний для сприйняття.

Для візуалізації типографіки використано шрифт без засічок (наприклад, OpenSans або Roboto), що забезпечує хорошу читабельність на екранах будь-якого розміру.

Кнопки оформлені зі згладженими кутами, мають легку тінь (ефект підняття), а також плавне затемнення при наведенні курсора, що підсилює інтерактивність.

Мітки товарів реалізовані у вигляді кольорових бейджів-кнопок із чіткими межами, що зручно сприймаються візуально та легко натискаються;

Блоки товарів: при наведенні мають ефект підсвічування, що покращує зворотний зв'язок з користувачем і створює приємний візуальний ефект.

Загалом, стилізація спрямована на формування інтуїтивного та естетично привабливого інтерфейсу, що відповідає потребам сучасного користувача.

Інтерфейс веб-сайту розроблено з урахуванням адаптивності – здатності коректно відображатися на пристроях із різними розмірами екранів (настільні ПК, ноутбуки, планшети, смартфони).

Для досягнення цієї мети застосовувалися такі технології:

медіа-запити CSS (@media) – дозволяють змінювати структуру та оформлення елементів в залежності від ширини екрану;

гнучкі сітки (Flexbox, CSS Grid) – використовувалися для динамічного розміщення блоків, що автоматично адаптуються під розмір екрану;

відносні одиниці виміру (% , em, vw, vh) – забезпечують масштабованість і гнучкість верстки у різних умовах перегляду;

мобільне меню (бургер-меню) – реалізовано як адаптивне випадаюче меню з плавною анімацією;

зображення – змінюють розмір відповідно до ширини контейнера, використовуючи властивості max-width: 100% та height: auto;

товарні блоки – при звуженні екрана переходять із горизонтального у вертикальний вигляд для кращої читабельності.

Завдяки цим рішенням сайт залишається функціональним, привабливим і зручним для користувачів на будь-яких пристроях. Інтерфейс містить горизонтальне меню з

основними розділами сайту: «Про сайт», «Ноутбуки», «Системні блоки», «Принтери» та «Форум». Центральну частину займає інформаційний блок з гаслом сайту, кнопками для вибору пристроїв і графічним елементом. На рис. 3.4 зображено головну сторінку веб-блогу, відкриту на персональному комп'ютері.

Рисунок 3.4 – Вигляд графічного інтерфейсу на персональному комп'ютері.

Інтерфейс сайту спроектовано максимально зрозуміло для користувача. Особливості UX-рішень включають:

чітко виділені кнопки дій (наприклад, «Перейти до магазину», «Порівняти ціни»), які розміщено безпосередньо під описом товару для зручності;

пошуковий рядок із кнопкою у вигляді піктограми-лупи, яка має анімований ефект наведення (hover);

зонування сторінки – усі розділи логічно структуровані, що дає змогу користувачеві швидко орієнтуватися на сайті;

товарні картки – містять зображення, назву, короткий опис, ціну та активне посилання, що відповідає стандартам сучасних інтерфейсів електронної комерції.

Такий вигляд інтерфейсу демонструє зручність навігації, чітку структуру й візуальну привабливість ресурсу.

На рис. 3.5 зображено головну сторінку веб-блогу, відкриту на смартфоні.

Рисунок 3.5 – Вигляд графічного інтерфейсу на смартфоні

### 3.2.2 Тестування функціональних елементів веб-блогу

Окремо перевірено функціонування форуму, де користувачі можуть залишати коментарі та обговорювати цифрову техніку. Усі елементи працюють стабільно та відповідають вимогам до функціональності веб-ресурсу.

На рис. 3.6 показано, як реалізовано функціонал переходу користувача від картки товару до сторінки магазину.

Рисунок 3.6 – Перевірка кнопок переходу до магазинів

Кнопки «детальніше» відкривають сторінку з описом пристрою та активними посиланнями на зовнішні торгові платформи. Це дозволяє швидко порівняти ціни та перейти до придбання обраної цифрової техніки.

На дуже малих екранах кнопки погано відображались. Було змінено CSS, щоб усе

працювало. Також перевірено швидкість сайту через PageSpeedInsights. Результат – 91 із 100. Це добрий показник.

Останнім етапом розробки стало тестування функціональних елементів веб-блогу. Це необхідно для того, щоб забезпечити зручність та стабільну роботу сайту для користувачів. У процесі перевірки було протестовано коректну роботу меню навігації, що відповідає за швидкий доступ до розділів блогу. Також перевірено відображення товарів на сторінках, включаючи зображення, описи та ціни. Особливу увагу приділено перевірці кнопок переходу до сторінок магазинів, які мають направляти користувача до зовнішніх ресурсів із технікою. Було протестовано, як сайт адаптується до різних розмірів екранів – зокрема, на смартфонах, планшетах та настільних комп'ютерах.

Також планується додати фільтри за цінами, рейтингами і брендами. Після тестування сайт буде опублікований на хостингу.

# Посилання

---

Це джерела виділених збігів у вашому документі. Кожен збіг позначено темно-зеленим числом, яке відповідає вказаному тут джерелу. Джерела впорядковані за схожістю — чим вищий бал, тим сильніше збіг.

#	Джерело	%
1	duikt.edu.ua	0.7%
2	theses.oa.edu.ua	0.3%
3	ibn.idsi.md	0.3%
4	dobest.it	0.3%
5	konf1.zzz.com.ua	0.3%
6	krypton.com.ua	0.3%
7	dobest.it	0.2%
8	xn----7sbbaqhlkm9ah9aiq.net	0.2%
9	dspace.znu.edu.ua	0.2%
10	elar.khmnu.edu.ua	0.2%
11	krs.chmnu.edu.ua	0.2%
12	er.nau.edu.ua	0.2%
13	ir.lib.vntu.edu.ua	0.2%
14	ela.kpi.ua	0.2%
15	stfalcon.com	0.2%
16	dspace.znu.edu.ua	0.2%
17	duikt.edu.ua	0.2%
18	metod.vntu.edu.ua	0.2%
19	dspace.nau.edu.ua	0.1%
20	css.in.ua	0.1%
21	elartu.tntu.edu.ua	0.1%
22	ela.kpi.ua	0.1%
23	ir.nmu.org.ua	0.1%

#	Джерело	%
24	core.ac.uk	0.1%
25	speka.media	0.1%
26	previous.scientia.report	0.1%
27	ekmair.ukma.edu.ua	0.1%
28	sea.gov.ua	0.1%
29	mediacom.com.ua	0.1%
30	krs.chmnu.edu.ua	0.1%
31	ela.kpi.ua	0.1%
32	unite.ai	0.1%
33	openarchive.nure.ua	0.1%
34	blog.depositphotos.com	0.1%
35	kart.edu.ua	0.1%
36	biblio.umsf.dp.ua	0.1%
37	lib.lntu.edu.ua	0.1%
38	dspace.wunu.edu.ua	0.1%
39	djinni.co	0.1%
40	labmv.knutd.edu.ua	0.1%
41	um.co.ua	0.1%
42	ua-referat.com	0.1%
43	copy-store.com.ua	0.1%
44	webcraftingcode.com	0.1%
45	moodle2.gi.edu.ua	0.1%



Дякуємо, що перевірили  
свій документ за допомогою  
Plag!