Engineer

HTML

мова, якою записано структуру сторінки

Що це таке

HTML, скорочення від HyperText Markup Language, яке досі часто називають HTML5, є простим текстовим форматом, у якому надходить кожна вебсторінка: це вміст, обгорнутий у теги, що кажуть, чим є кожна частина, щоб браузер міг відрізнити заголовок від абзацу, посилання від пункту списку, а поле форми від його підпису.

Браузер читає цей текст згори донизу й будує з нього дерево елементів, Document Object Model, або DOM. Саме це дерево оформлюють таблиці стилів, змінюють скрипти й озвучують екранні читачі. Розподіл праці у вебі давній і досі тримається: HTML каже, чим щось є, CSS каже, як воно виглядає, а JavaScript каже, що воно робить, коли його торкаються. Елемент на кшталт `<nav>`, `<button>` чи `<h2>` несе значення, з яким може працювати кожен із цих читачів. Справжній елемент `<button>` отримує фокус клавіатури, роль, яку оголошують допоміжні технології, і спрацьовує від Enter або пробілу без жодного рядка додаткового коду.

Мова починалася як спосіб семантично описувати наукові документи, а виросла в шар, що лежить під майже кожним застосунком із вебінтерфейсом. Перші п’ять років, 1990–1995, вона розвивалася в CERN, а потім в IETF; HTML 3.2 завершили у W3C 1997 року, а невдовзі з’явився HTML4. Роками W3C і WHATWG вели окремі тексти, що дедалі більше розходилися, аж поки 2019 року не домовилися працювати над одним: HTML Living Standard, який підтримує WHATWG і реалізують усі основні браузерні рушії. Номера версії він не має. На запитання «це HTML5?» стандарт коротко відповідає «так» і називає цей термін модним словом для сучасної вебплатформи, а не назвою випуску.

Де HTML стоїть поруч із тим, із чим його найчастіше порівнюють? Markdown є скороченим записом, який інструмент збирання перетворює на HTML. PDF прив’язує макет до друкованої сторінки, тоді як HTML перетікає під будь-який екран, на який потрапляє. XHTML є тим самим словником, записаним за суворішими правилами XML. Шаблони JSX, Jinja і Hugo є способами створювати HTML, і те, що від кожного з них доходить до браузера, однаково є HTML. Команда, яка зважує сайт ручної роботи проти конструктора сайтів, обирає, хто писатиме розмітку, а не те, чи буде вона взагалі.

Застосуйте поняття до рішення

Застосуйте «Що таке HTML?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Інтерфейси, що швидко завантажуються, працюють на телефоні й поводяться однаково в усіх браузерах ваших клієнтів.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Керувала full-stack-розробкою GIS-карт — full-stack-картографія — від бази даних до браузера

Керувала full-stack-розробкою GIS-карт Зв'язатися з нами →

Чому це важливо

Мало хто обирає HTML свідомо. Він з’являється таким, яким його видає фреймворк, конструктор сторінок чи система керування вмістом. Справжній вибір полягає в тому, чи буде ця розмітка частиною роботи, яку хтось перевіряє, чи побічним продуктом, якого ніхто не читає, і наслідки цього вибору проявляються в трьох місцях.

Перше: хто зможе користуватися сторінкою. Допоміжні технології читають дерево, а не дизайн. Користувач екранного читача рухається сторінкою за заголовками й орієнтирами; користувач клавіатури переходить табуляцією від одного посилання чи елемента керування до наступного. Стилізований `<div>` з обробником кліку виглядає як кнопка для зрячого користувача миші і є звичайним текстом для двох інших. Вимоги доступності, як-от WCAG, здебільшого сформульовано щодо того, що відкриває розмітка, тож виконати їх після запуску зазвичай означає переписати шаблони, а не підправити кольори.

Друге: як довго проживе вміст. Браузери досі відкривають сторінки, написані в 1990-х, бо зворотна сумісність є одним із принципів, за якими підтримують стандарт. Фреймворки змінюються що кілька років. Вміст у чистому, валідному HTML можна перенести в іншу систему, заархівувати,

конвертувати в інші формати чи переоформити, не переписуючи. Вміст, похований у вкладених обгортках конструктора сторінок, зазвичай доводиться копіювати вручну.

Третє: швидкість і охоплення. Сторінку, яку сервер надсилає повністю, можна малювати в міру надходження байтів. Застосунок, що надсилає майже порожню оболонку й будує документ у JavaScript, змушує кожного читача чекати на скрипт, а все, що читає сторінку без його виконання, від попереднього перегляду посилання до багатьох краулерів, бачить лише оболонку. Для продукту за логіном, де суть саме у взаємодії, такий обмін може бути виправданим. Для маркетингових сторінок, документації й усього, що мають знаходити через пошук, готовий HTML є дешевшим варіантом за замовчуванням.

Ціна всіх трьох однакова й невелика: людина в команді, яка читає HTML так само уважно, як код. Семантична розмітка майже нічого не коштує, коли її пишуть одразу, і коштує дуже дорого, коли її додають заднім числом.

Як це виглядає на практиці

Найпростіший приклад під рукою дає цей сайт. Його сторінка подяк перелічує, що доходить до браузера відвідувача: HTML, одна таблиця стилів, локальні вебшрифти, зображення, без виконуваного JavaScript. Кожну сторінку повністю записано як файл на диску ще до того, як її хтось запросить, а не зібрано з шаблону на кожен запит. Шлюз якості сайту перевіряє цей HTML за допомогою vnu і вважає кожне повідомлення помилкою, а та сама сторінка посилається на валідатор розмітки W3C, який використовує той самий рушій для перевірки відповідності живому стандарту, тож читач може перевірити твердження сам.

Попередня клієнтська робота підходила до HTML з іншого боку. У тридцяти вебсайтах, побудованих за готовими візуальними дизайнами файли Photoshop перетворено на HTML вручну. Стандарт мав дві сторони: точність щодо відступів, типографіки й деталей, які мав на увазі дизайнер, і гнучкість, достатню, щоб сайт працював і тоді, коли хтось уперше редагував його вміст. Збіг із макетом становить половину роботи; розмітка, що переживає наступну правку, становить другу половину.

Збереження HTML як канонічної форми також залишає можливість публікувати поруч інші формати. Коли сайт віддзеркалили в Gemini і Gopher, те саме збирання на зафіксованому етапі реалізації створило 777 документів Gemini і 777 меню Gopher з того самого джерела. Окреме порівняння збирань показало зміну HTML на нуль байтів під час додавання цих форматів; цей результат не описує пізніших редакторських чи візуальних змін.

Де виникають проблеми

Макет зсувається так, що жодна таблиця стилів цього не пояснює. Сторінка, що починається без `<!DOCTYPE html>` або зі старого й хибного доктайпу, вмикає в браузері режим сумісності (quirks mode), який імітує старі браузери. Відсоткові висоти, комірки таблиць і висота рядків тоді поводяться по-старому, відступаючи від специфікацій CSS, і результат відрізняється від тієї самої сторінки в стандартному режимі. Виправлення займає один рядок на початку кожного шаблону.

Зламана розмітка відображається, але дерево виходить іншим. Парсер HTML відновлюється після кожної помилки визначеним способом, однаково в усіх браузерах, тож помилки рідко видно на екрані. Вони спливають пізніше, коли селектор чи скрипт шукає елемент не там, куди його поставив парсер. Класичний випадок: блоковий елемент усередині абзацу.

```
<p>Перший рядок
  <div>Блок, вкладений усередину</div>
</p>
```

Парсер закриває абзац перед `<div>`, а зайвий `</p>` перетворює на другий абзац без вмісту. Запускайте валідатор під час збирання й зупиняйте збирання на його помилках; терпимість браузера є механізмом відновлення, а не перевіркою.

Кожен елемент керування стає `div`. Клікабельні елементи `div` і `span`, заголовки, обрані за розміром, нижній індекс, використаний, щоб зробити текст меншим: кожен із цих прийомів виглядає правильно й повідомляє допоміжним технологіям неправду. Користувачі клавіатури проходять повз елементи керування, які ніколи не отримують фокусу, а користувачі екранних читачів втрачають заголовки й орієнтири, за якими

переміщаються. Сторінка без елемента `<main>` чи `<h1>` змушує їх щоразу читати з самого початку. Використовуйте елемент, що називає завдання, як-от `button`, `a`, `h2`, `main` чи `nav`, а вигляд змінюйте через CSS.

Сторінка лишається порожньою, доки не виконається скрипт. Сайт із рендерингом на клієнті надсилає мало що, крім `<div id="app"></div>`, і будує документ у браузері. На повільному телефоні читач чекає; якщо скрипт падає або його блокує фільтр, сторінка так і лишається порожньою. Рендерте документ на сервері або під час збирання, а взаємодію додавайте поверх HTML, який уже має сенс сам по собі.

Що почитати далі

- HTML Living Standard — специфікація, яку реалізують браузери; її підтримує WHATWG і постійно оновлює.
- Introduction — історія від CERN до угоди 2019 року і власна відповідь стандарту на запитання про HTML5.
- Parsing HTML documents — відновлення після помилок, спільне для всіх браузерів, розписане правило за правилом.
- The HTML syntax — доктайп, елементи й атрибути, а також те, що робить документ відповідним стандарту.
- The Nu Html Checker — рушій, на якому працюють vnu і валідатор розмітки W3C, з інструкцією для командного рядка.
- DOM Standard — дерево, яке будує парсер, та інтерфейси, якими скрипти його змінюють.

Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке HTML?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/html/