Engineer

HTML

ページの構造を記述する言語

どのようなものか

HTMLはHyperText Markup Languageの略で、今も広くHTML5と呼ばれます。すべてのウェブページが届くプレーンテキスト形式です。内容を、その部分が何なのかを示すタグで囲むため、ブラウザーは見出しと段落、リンクとリスト項目、フォームの入力欄とそのラベルを区別できます。

ブラウザーは、そのテキストを上から下へ読み、要素の木構造であるDocument Object Modelを作ります。スタイルシートが装飾し、スクリプトが変更し、スクリーンリーダーが読み上げるのは、その木です。ウェブでの役割分担は古くから変わりません。HTMLは対象が何であるか、CSSはどう見えるか、JavaScriptは操作時にどう動くかを記します。`<nav>`、`<button>`、`<h2>`などの要素には、これらすべての読み手が利用できる意味があります。本物の`<button>`には、追加コードを一行も書かずに、キーボードのフォーカス、支援技術が読み上げる役割、EnterやSpaceによる実行が備わります。

科学文書を意味に基づいて記述するために始まった言語は、ウェブ画面を持つほぼすべてのアプリケーションを支える層へ育ちました。最初の五年間、1990–1995年は、CERN、次いでIETFで開発されました。HTML 3.2は1997年にW3Cで完成し、間もなくHTML4が続きました。長年、W3CとWHATWGは食い違う仕様を維持していましたが、2019年、WHATWGが保守し、すべての主要なブラウザーエンジンが実装するHTML Living Standardへ一本化する合意を結びました。バージョン番号はありません。「これはHTML5か」という問いに、仕様は短く肯定し、その言葉をリリース名ではなく、現代のウェブ基盤を指す流行語と呼びます。

比較されるほかのものとは、どんな関係でしょうか。Markdownは、ビルドツールがHTMLへ変換する簡略記法です。PDFが印刷ページにレイアウトを固定するのに対し、HTMLは表示先の画面に合わせて流れを組み直します。XHTMLは、同じ語彙を、より厳しいXMLの規則で記述したものです。JSX、Jinja、Hugoのテン

プレートはHTMLを生成する方法であり、いずれからブラウザーへ届くものもHTMLです。手作りのサイトとウェブサイト作成ツールを比較するチームが選ぶのは、誰がマークアップを書くかであり、マークアップの有無ではありません。

概念を判断に生かす

「HTMLとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 読み込みが速く、スマートフォンでも使え、お客様がどのブラウザーを使っていても同じように動く画面を提供します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 フルスタックのGIS地図開発を指揮 — DBからブラウザーまでのフルスタック地図開発

フルスタックのGIS地図開発を指揮 お問い合わせ →

なぜ重要なのか

HTMLを意識して選ぶチームは、多くありません。フレームワーク、ページ作成ツール、コンテンツシステムが出力したものとして届きます。実際に選んでいるのは、その出力を誰かが検査する納品物とするか、誰も読まない副産物とするかです。影響は三つの面に現れます。

第一は、誰がページを使えるかです。支援技術が読むのは、デザインではなく木構造です。スクリーンリーダーの利用者は見出しやランドマークを使って移動し、キーボードの利用者はTabキーでリンクや操作部品を順に移ります。クリック処理を付けて装飾した `<div>` は、目で見てマウスを使う人にはボタンに見えますが、スクリーンリーダーとキーボードの利用者にとっては、ただのテキストです。WCAGなどのアクセシビリティ要件は、主にマークアップが公開する情報に基づいて書かれているため、公開後に満たそうとすると、色の調整よりもテンプレートの書き直しになりがちです。

第二は、コンテンツがどれだけ長く使えるかです。後方互換性が仕様の保守原則の一つであるため、ブラウザーは、1990年代に書かれたページも今なお開けます。フレームワークは数年ごとに入れ替わります。整理された妥当なHTMLに収めたコンテンツなら、書き直さずに別システムへ移し、保存し、ほかの形式へ変換し、デザインを変えられます。ページ作成ツールの入れ子のラッパーに埋もれたコンテンツは、通常、手で取り出す必要があります。

第三は、速さと届く範囲です。サーバーが完成した状態で送るページは、バイトが届くにつれて描画できます。ほぼ空の外枠を送り、JavaScriptで文書を組み立てるアプリケーションは、すべての読者をスクリプト待ちにし、リンクプレビューから多くのクローラーまで、それを実行せずに読むものには、外枠しか見せま

せん。操作が目的の、ログイン後に使う製品では、そのトレードオフを受け入れる価値がある場合もあります。マーケティング用ページ、ドキュメント、検索で見つけてもらうものでは、完成したHTMLを基本とするほうが低コストです。

この三つに必要な費用は同じで、小さいものです。コードと同じ慎重さでHTMLを読む人を、チームに置くことです。意味に合ったマークアップは、初めから書けばほぼ費用がかからず、後付けすれば大きな費用がかかります。

実際の活用例

このサイトが、最も身近で単純な例です。クレジットのページは、閲覧者のブラウザーへ届くものを列挙します。HTML、スタイルシート一つ、ローカルのウェブフォント、画像であり、実行可能なJavaScriptはありません。すべてのページは、要求のたびにテンプレートから組み立てるのではなく、誰かが要求する前に、ディスク上のファイルへ全文を書き出します。サイトの品質ゲートは、vnuでそのHTMLを検証し、すべての診断をエラーとします。同じページには、このエンジンをLiving Standardに対して実行するW3C Markup Validatorへのリンクもあるため、読者は説明を信じるだけでなく、自分で確かめられます。

以前の顧客向け業務では、逆方向からHTMLへ取り組みました。完成したビジュアルデザインから構築した三十のウェブサイトでは、Photoshopファイルを手作業でHTMLへ変換しました。基準は二つありました。デザイナーが意図した間隔、書体、細部への忠実さと、初めてコンテンツを編集したときにもサイトが動き続ける柔軟さです。見本に合わせるのが仕事の半分で、次の編集に耐えるマークアップが、もう半分です。

HTMLを正本とすることで、別の形式も並べて公開できます。サイトをGeminiとGopherへ複製したとき、記録された実装段階では、一つのビルドが同じ情報源から、777件のGemini文書と777件のGopherメニューを生成しました。別のビルド比較では、それらの形式を追加したとき、HTMLの変化がゼロバイトと測定されました。この結果は、その後の編集や見た目の変更を述べたものではありません。

問題が起きるところ

スタイルシートで説明できないレイアウトのずれが起きます。 先頭に `<!DOCTYPE html>` がないか、古く不正なものがあるページは、古いブラウザーを模倣する描画方式、後方互換モードへ入ります。パーセント指定の高さ、表のセル、行の高さが、CSS仕様とは異なる古い動作に従い、標準モードの同じページとは結果が変わります。修正は、各テンプレートの先頭の一行です。

壊れたマークアップも描画されますが、木構造が変わります。 HTMLパーサーは、定義された方法で、すべてのブラウザーで同じように、あらゆる誤りから回復するため、間違いが画面へ現れることはほとんどありません。後で、セレクターやスクリプトが、パーサーの置いた場所とは違う場所で要素を探したときに表面化します。典型例は、段落の中にブロック要素を置く場合です。

```
<p>最初の行
  <div>中に置いたブロック</div>
</p>
```

パーサーは、`<div>` の手前で段落を閉じ、余った `</p>` を、何も入っていない二つ目の段落へ変えます。ビルドで検証器を実行し、エラーがあれば失敗させます。ブラウザーの寛容さは回復機能であり、内容を検査しているわけではありません。

操作部品が、すべて `div` になります。 クリックできる `div` や `span` 、大きさで選んだ見出し、文字を小さくするための下付き文字は、どれも正しく見えながら、支援技術へ誤った意味を伝えます。キーボードの利用者は、フォーカスを受けない操作部品を通り過ぎ、スクリーンリーダーの利用者は、移動に使う見出しやランドマークを失います。 `<main>` 要素や `<h1>` がないページでは、毎回先頭から読むことになります。 `button` 、 `a` 、 `h2` 、 `main` 、 `nav` など、役割を表す要素を使い、見た目はCSSで変えます。

スクリプトが動くまで、ページが空白です。 クライアント描画のサイトは、ほぼ `<div id="app"></div>` だけを送り、ブラウザーで文書を組み立てます。処理の遅いスマートフォンでは読者が待ち、スクリプトでエラーが起きるか、フィルターに遮断されると、ページはずっと空白のままです。サーバーかビルド時に文書を描画し、それだけでも意味が通るHTMLの上へ、操作機能を追加します。

参考資料

- HTML Living Standard — WHATWGが保守し、継続的に更新する、ブラウザーが実装する仕様です。
- HTML標準の紹介 — CERNから2019年の合意までの歴史と、HTML5という問いへの仕様自身の回答です。
- HTML文書の解析 — すべてのブラウザーが共有する誤りからの回復を、規則ごとに記したものです。
- HTMLの構文 — doctype、要素、属性、適合する文書の条件です。

- Nu Html Checker — vnuとW3Cのマークアップ検証器を支えるエンジンと、そのコマンドラインでの使い方です。
- DOM標準 — パーサーが作る木構造と、スクリプトが変更に使うインターフェースです。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「HTMLとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/html/