

Atlas-Ready Website Developer Checklist

Ensure your website is understandable by the Atlas ChatGPT browser and future AI retrieval engines.

1. Semantic HTML

- ☐ Use `<main>`, `<article>`, `<section>`, `<aside>`, `<header>`, `<footer>`, `<nav>` correctly.
- ☐ Maintain a clear heading hierarchy (`<h1>` → `<h2>` → `<h3>`).
- ☐ Avoid empty or meaningless `<div>` wrappers.
- ☐ Use semantic lists (`<ul>`, `<ol>`, `<dl>`) for structured data.
- ☐ Use `<table>` for data, not layout.

2. Structured Data (Schema.org)

- ☐ Add JSON-LD for key entities:
 - `Organization`
 - `Person`
 - `Article / BlogPosting`
 - `Event`
 - `Product`
 - `BreadcrumbList`
- ☐ Validate with validator.schema.org.
- ☐ Include `@id` references and canonical URLs.

3. Metadata and Linking

- ☐ `<title>` and `<meta name="description">` accurately summarise each page.
- ☐ Include `og:` and `twitter:` tags for preview context.
- ☐ Add `<link rel="canonical">` to prevent duplicates.
- ☐ Include `rel="alternate"` for multilingual pages.
- ☐ Ensure `<link rel="author">` or `<meta name="author">` present where relevant.

4. Content Accessibility

- ☐ All images have descriptive `alt` attributes.
- ☐ Forms include `<label>` or `aria-label`.
- ☐ Links are descriptive ("Download the guide" not "Click here").
- ☐ Keyboard navigation and focus order tested.
- ☐ Use skip links and proper heading structure.

5. Content Visibility

- ☐ Core content is rendered in initial HTML (not JS only).
- ☐ Server-Side Rendering (SSR) or Static Site Generation (SSG) used for React/Vue.

- ☐ `<noscript>` fallback for essential content.
- ☐ Links use real `<a href>` elements (no JS-only navigation).

6. Performance & Stability

- ☐ LCP < 2.5 s
- ☐ CLS < 0.1
- ☐ INP < 200 ms
- ☐ Use lightweight, non-blocking assets.
- ☐ Preload key fonts and hero images.

7. Language & Readability

- ☐ Write in clear, factual sentences.
- ☐ Use consistent entity names (e.g., “Science Council” not “it”).
- ☐ Include summaries or key takeaways per page.
- ☐ Break long content into short paragraphs.

8. Retrieval Optimisation

- ☐ Provide an `/api/` or JSON feed for content where possible.
- ☐ Consider `.well-known/ai-plugin.json` or OpenAPI spec for data endpoints.
- ☐ Ensure sitemap.xml is up to date and crawlable.
- ☐ Use RSS/Atom feeds for blogs and updates.

9. Accessibility as Machine Context

- ☐ Ensure WCAG 2.2 AA compliance.
- ☐ Test with screen readers (NVDA, VoiceOver).
- ☐ Provide captions or transcripts for video/audio.

10. Validation & Testing

- ☐ Run `curl -s https://yourdomain.com | htmlq -t` to inspect raw text output.
- ☐ Paste raw HTML into ChatGPT and ask: “What does this page describe?”
- ☐ Fix ambiguous or missing semantics until the AI returns accurate summaries.

Optional: WordPress-Specific

- ☐ Use core blocks (`core/heading`, `core/paragraph`, `core/list`) for clarity.
- ☐ ACF fields should output real text, not placeholders.
- ☐ Register schema data via `wp_head` hook.
- ☐ Confirm permalinks and canonical URLs are correct.
- ☐ Avoid JS-only content in `render_block` filters.