

# Is your website usable by AI?

*It's not just about being found – it's about being usable.*



# Hi, I'm Lisa

I've been making websites for over 20 years and run an award-winning web development agency based just outside Bristol in the UK. You can find [our website here](#).

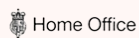
I'm a speaker at events, a writer of all things web, and have taught SEO for longer than I care to remember. I work with clients all over the world, from small one-person-bands to household names. My clients include...

I also founded a not-for-profit to help people find the best web developer for their project, rather than fall foul of a smooth-talking cowboy. From that role, I was asked to represent the industry in discussions with the EU about online privacy reform.

[Instagram](#) [LinkedIn](#)



Sainsbury's



THE AGENTIC WEB SERIES

# Four sessions, one theme...

May      Why AI keeps choosing the same brands - *replay available*

---

June      The new technical SEO: what actually matters now – *replay available*

---

**July**      The agentic web: is your website usable by AI? - *today*

---

September      Selling and building websites in 2027

# Where the series has got to

Three sessions, three increasingly big asks of your website.

May — Will AI TRUST you? (continuous due diligence)

June — Can AI READ you? (the new technical SEO, entity data)

**July — Can AI USE you? (today)**

*For twenty years we built the web to keep  
(most) robots OUT.*

*Now we need to let some of them back in - the  
ones working for your customer.*

*Most websites are still blocking everyone.*



# This isn't 5 years away...

- Agentic browsing already exists - AI that drives a real browser, reads the screen, moves the cursor and completes tasks.
- The big AI tools have shipped agent / operator modes, and the shopping-agent direction has already been tried.
- People are already comfortable asking AI to “just sort it out” - and the obvious things to hand off are quite likely the things your site is for.

**SO THE QUESTION ISN'T “WHETHER”**

It's not whether an AI agent will try to use your site. It's whether it'll SUCCEED when it does.

Jan 12, 2026

**BUSINESS INSIDER**[Subscribe](#)

Today's Briefing  **BREAKING** SpaceX Stock Stumbles | IBM's AI Warning | Boomer Bailout

AI

# McKinsey CEO Bob Sternfels says the firm now has 60,000 employees: 25,000 of them are AI agents

By [Lakshmi Varanasi](#) [+ Follow](#)



# Just to be clear: what is an agent?

A chatbot **ANSWERS**. An agent **DOES**.

It takes a goal - “book me a plumber for Tuesday”, “re-order my usual” - and goes off and completes the multi-step task, using a mix of AI reasoning, your APIs, and whatever it can read on the open web. So instead of a person clicking through your site, software is doing it - on that person’s behalf, to finish a job they’ve handed off.



PART ONE

# How an AI agent actually “sees” your site

Get this right and everything practical afterwards makes sense.

# Three ways a machine can use your site

1. It reads the CODE - the HTML, the structure, the labels underneath. (This is largely what June was about.)
2. It looks at the SCREEN - literally screenshots the page and works out what to do from the picture, the way a person would.
3. It connects programmatically – the systems “talk” to each other via API or MCP.

**Modern agents do ALL.**

So from here on, you're building for two AI readers at once - the one reading your code or using a MCP/accessing you site programmatically, and the one looking at your page.

# A new bit: AI looks at the picture

Google has talked openly about AI that understands a page VISUALLY - taking a screenshot and interpreting the layout like a human: where's the button, what does this look like it does, what's the main action here.

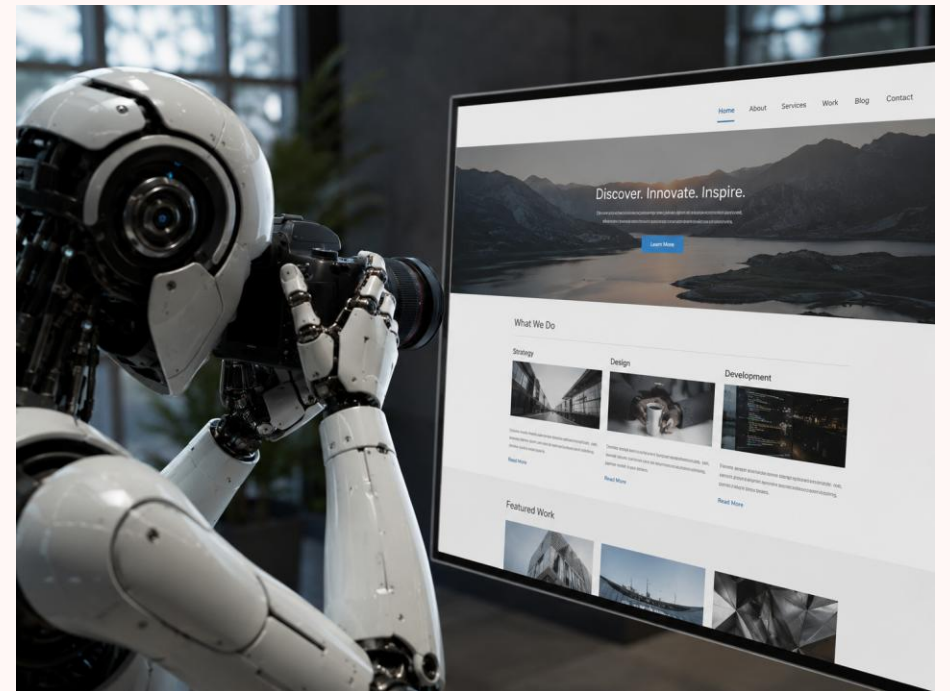
**Which means some things suddenly matter that SEO never cared about:**

**What's in view immediately?**

**Is that button clearly designed? Is there even a button?!**

**Is the font legible?**

**Blog post: 5 things we're doing differently on website designs now**



# Why layout shift suddenly matters more

## In plain English:

If an agent screenshots your page to decide where to click, and things are still jumping around as it loads - an image pops in, a cookie banner shoves everything down - it can misjudge where the button is. Or click the wrong thing entirely.

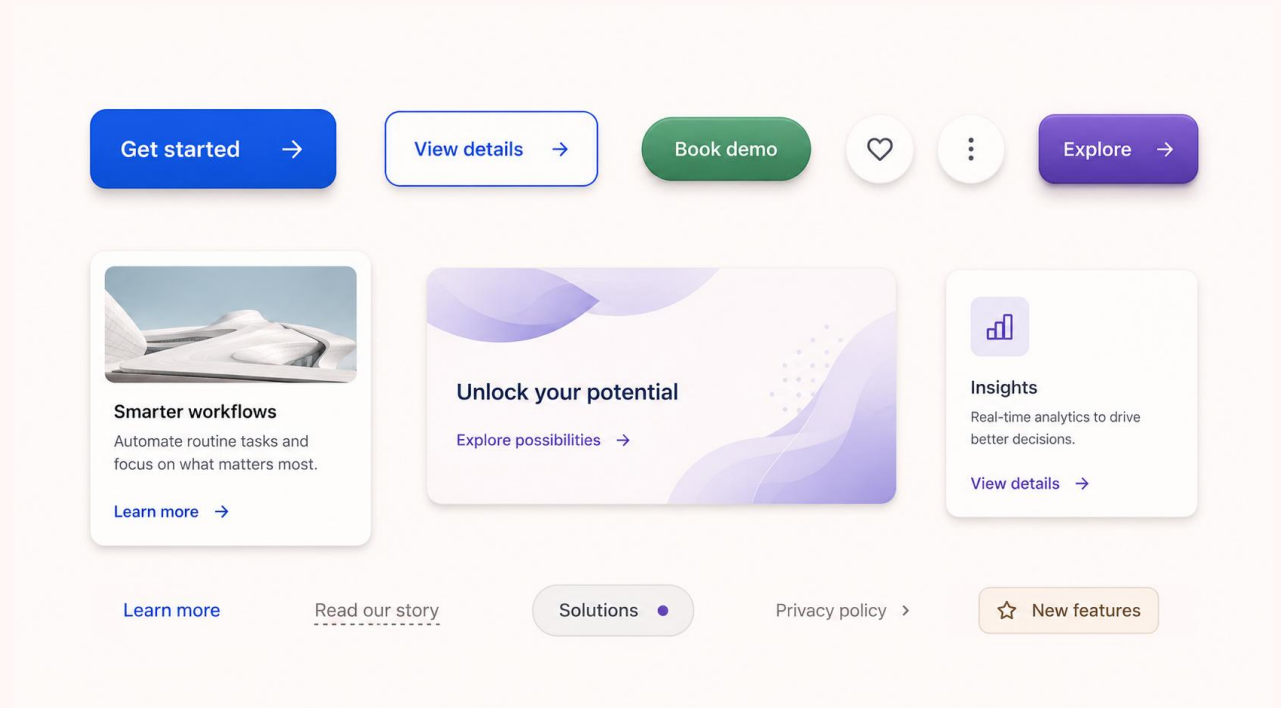
### ⚙ THE TECHNICAL BIT

This is CLS - Cumulative Layout Shift - one of Google's Core Web Vitals: a score for how much stuff moves around as a page loads. It used to be a UX-and-ranking metric. Now it's also a "can the agent hit the right target" metric. You can check it free in [Google's PageSpeed Insights](#) or in a [free SiteVitals audit](#) (which includes lots of other checks too).

# Buttons need to look like buttons

If an agent is reading the screen, a thing that's STYLED to look like a button but isn't coded as one is ambiguous - it could be a button, a banner, or just decoration. A human guesses right from context. An agent might not.

**So the rule that runs through this whole talk: your page's LOOK and its CODE have to agree. When they don't, that's where agents fall over.**



# Slideshows and hidden text

Traditional search engine robots read code, and all the code (of a SEO'd site) is viewable in the source code, so it doesn't matter how it's presented on the front end.

**BUT NOW** – if agents are using screenshots, stuff that's hidden is just, well – hidden.



## Web Design & Development

## Strategy & Consultancy

## Architecture & Hosting

We can help you piece together the perfect digital architecture, no matter your traffic levels and requirements. We also then have advanced inhouse tooling to monitor everything for you.

If you're bursting at the seams of your existing hosting and want to explore solutions like auto-scaling or self healing server set ups with AWS or DigitalOcean, we'll get you up and running in no time.

[Learn more](#) ↗

emotionally invested in our clients' success, and we're really nice people to work with.

**The result?** Cost-effective web development, pragmatic strategy, rapid business progression, and relationships that last for decades.

# 36M

people visit the websites we manage every year.

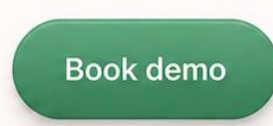
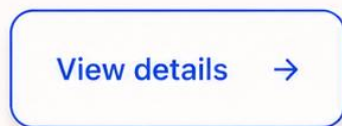


# Actionable HTML

When an agent reads the code, it needs the markup to say not just what something **IS**, but what it **DOES**.

That idea - “actionable HTML” - is turning up in Google’s own documentation. A button labelled “button” tells it nothing.

**A button labelled (in the code and to users) “Add to basket” or “Book a discovery call” tells it exactly what will happen if it acts.**



**Whether an agent reads your screen or your code, it's asking the same thing a human asks: *“what can I do here, and what happens if I do it?”***

*Your job is to make that answerable without guessing - in the picture AND in the code, with the two agreeing.*

## PART TWO

# The actions on your site - and whether a machine can use them

# First: which actions actually matter?

Every site has a handful of actions that earn their keep. You need to make sure an agent can do them otherwise there's no point going to lots of effort to get them there.

For a SHOP: [search](#) · [filter](#) · [choose a variant](#) · [add to basket](#) · [checkout](#)

For a SERVICE business: [book a call](#) · [request a quote](#) · [fill the contact form](#) · [download the thing](#)

Find your two or three make-or-break actions. Those are the ones an agent has to be able to complete.



# A button should actually be a button

## ✗ THE WAY THAT TRIPS AGENTS UP

```
<div class="btn"  
  onclick="addToCart()">  
  Add to basket  
</div>
```

*Looks like a button. To an agent reading the code OR the screen, it's a mystery box - might be clickable, might not.*

## ✓ THE WAY AN AGENT UNDERSTANDS

```
<button type="button"  
  onclick="addToCart()">  
  Add to basket  
</button>
```

*Actually a button. An agent knows it can click it - and the label says what will happen.*

Identical to a human. Completely different to a machine.

# Say what the action does, not “click here”.

“Click here”, “Submit” and “Learn more” tell an agent nothing about what happens next. Specific labels tell it everything.

- “Click here” → “Download the pricing guide”
- “Submit” → “Book your free call”
- “Learn more” → “See how SiteVitals works”

Same fix helps screen-reader users too - which is the theme of this whole section, and we’ll come back to it.

# A form an agent can fill in

## ✗ THE WAY THAT TRIPS AGENTS UP

```
<span>Email</span>
<input type="text">

<span>Phone</span>
<input type="text">
```

*The labels just float nearby on screen. In the code, nothing connects them — the agent can't tell which box is which.*

## ✓ THE WAY AN AGENT UNDERSTANDS

```
<label for="email">Email</label>
<input id="email" name="email"
      type="email"
      autocomplete="email">

<label for="phone">Phone</label>
<input id="phone" name="phone"
      type="tel"  autocomplete="tel">
```

*Every field labelled, named and typed. The agent fills it correctly — and so does the customer's browser autofill.*

This is the difference between capturing the lead and losing it at the very last step.

# The good news - it's accessibility work

Almost everything in the code that makes your site usable by an AI agent is the SAME thing that makes it usable by someone with a screen reader: semantic HTML, real buttons, labelled forms, descriptive links, ARIA where it's needed. (ARIA - extra attributes that spell out what an element is and does.)

**So this isn't new work fighting for budget. It's the SAME work, with a second payoff.**

## THE EASY WAY TO SEE WHERE YOU STAND

SiteVitals checks for lots of accessibility items in it's SEO/AI visibility check - flagging the unlabelled buttons, the missing form labels, the structure problems. [Give it a go here >](#)

# Semantic structure: the map an agent navigates

## In plain English:

Semantic elements tell an agent which part of the page it's in and how to move around. A page built from nothing but identical generic boxes is a map with no street names - the agent can see shapes, but not what they mean.

### ⚙️ THE TECHNICAL BIT

Use the elements that carry meaning: `<nav>`, `<main>`, `<header>`, `<article>`, `<button>` - rather than a sea of `<div>` tags that say nothing about what they contain. This is invisible to your human visitors and hugely meaningful to a machine. It costs nothing at build time and is painful to retrofit - so it's a brief-stage decision, not a later one.



# Deep dive: e-commerce

For the shops in the room - the agentic checkout is coming. Go and test each of these.

- Variant selection (size, colour) hidden behind JavaScript an agent can't operate.
- "Add to basket" that's a styled `<div>` rather than a real button
- A checkout flow that breaks when you try to do it with a keyboard.
- Stock and price that only appear after scripts run - so the agent sees an empty or wrong value

If an AI is buying on a customer's behalf, every one of these is a point where the sale fails - and you'd never see it, because it works fine for you.

# Deep dive: service businesses

For the service businesses - your make-or-break action is usually “get in touch”. Same idea, different failure points.

- Contact forms with unlabelled fields the agent can't fill (see the code slide earlier)
- Booking widgets embedded so the agent can't see into them (more on third parties next)
- A “Book a call” that's a styled `<div>`, or a calendar that's pure JavaScript
- Multi-step enquiry flows that lose the agent halfway through

If an AI is helping someone find and book a consultant like you - can it actually book YOU? Or does it give up and pick the firm whose form it could fill?

# The bit you don't control: third-party widgets

Booking tools, payment, review widgets, chat, calendars - often your most important action lives in someone ELSE's code, embedded on your page.

**Ask the provider:** Is the widget keyboard and screen-reader accessible? Does its content work without running JavaScript? Is it embedded as an iframe an agent can't see into? Do you put a CAPTCHA on the booking step - and can that be turned off?

**Test it yourself:** try to complete the booking or payment using **ONLY** the keyboard (Tab to move, Enter to submit a form). If you can't get through it, an agent probably can't either.

## THE UNCOMFORTABLE TRUTH

You can build your own site perfectly and still hit a wall at the critical moment, because the booking lives in third-party software you don't control. You're only as usable as your weakest embedded tool.

# The next step... webMCP

## ✓ PERFECTLY FINE FOR AN AGENT

```
<label for="email">Email</label>
<input id="email" name="email"
      type="email"
      autocomplete="email">

<label for="phone">Phone</label>
<input id="phone" name="phone"
      type="tel" autocomplete="tel">
```

Instead of the agent scraping the form and simulating typing, WebMCP wraps that form in a JSON schema. The browser then tells the agent: *"Don't bother clicking or typing. Just send me a structured JSON payload containing {email: "xyz", phone: "123"} and I will execute the submission instantly."*

## ✓ THE WAY AN AGENT UNDERSTANDS

```
<form id="contactForm" toolname="send_message"
      tooldescription="Submit user contact details
      and inquiries">
```

```
<label for="email">Email</label> <input
type="email" id="email" name="email" required
toolparamdescription="The user's primary
contact email address">
```

```
<label for="tel">Phone (Optional)</label>
<input type="tel" id="tel" name="phone"
toolparamdescription="The user's phone number
including area code">
```

```
<button type="submit">Submit Details</button>
</form>
```

# Other things you might hear about...

**MCP**

**Tool & data access**

Lets an agent plug into your data and APIs to actually do things. (Started by Anthropic.)

**A2A**

**Agent-to-agent**

A common language for agents to talk to each other and hand work over.

**UCP**

**Agentic discovery & commerce**

Google-led. How an agent discovers what you sell and works through a purchase across surfaces.

**ACP**

**In-surface checkout**

OpenAI/Stripe. Running a checkout inside an AI surface. (The protocol persists; the first in-chat version didn't last very long.)

**AP2**

**Payment authorisation**

Google + the card networks. The cryptographic “the customer really did authorise this” layer.

**ⓧ402**

**Internet-native payments**

Checks requests for payment – if there's been no payment the server responds with HTTP 402, prompting the client to pay and retry.

**MPP**

**Machine to machine payments**

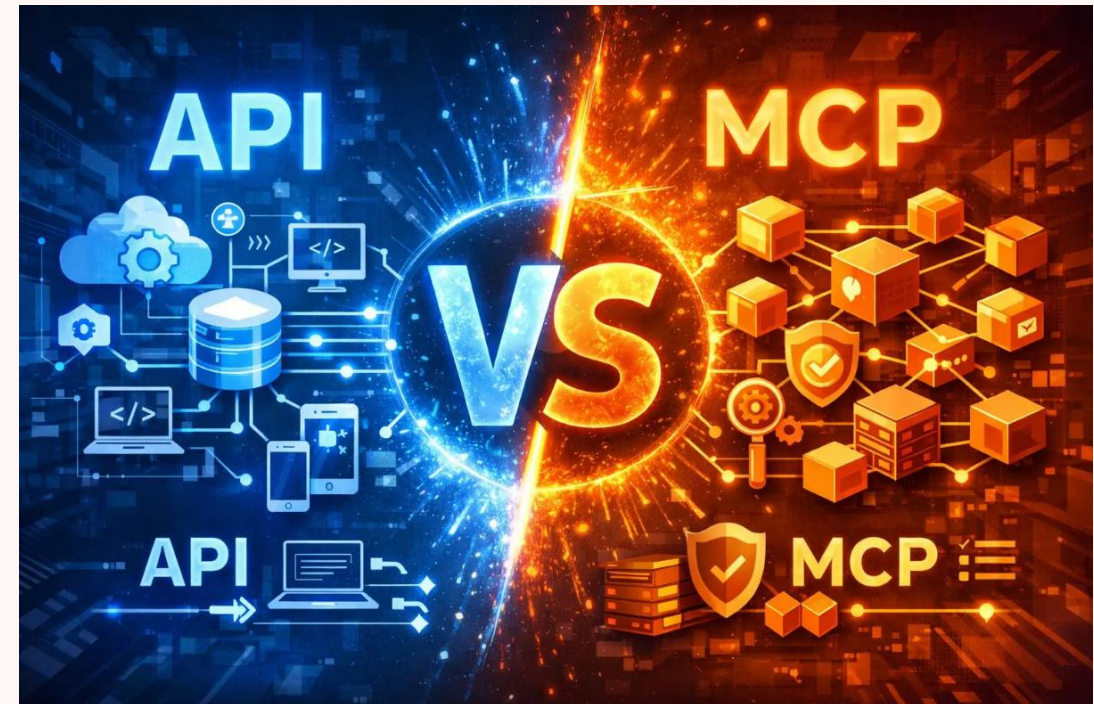
The open protocol for machine-to-machine payments.

# A little more on MCP...

Think of a traditional API as a custom blueprint. Every single app uses a different one, forcing human developers to build separate, custom bridges for an AI to use them.

MCP flips this script entirely. It acts like the universal USB-C plug for the AI era. It allows any software or website to instantly speak a single language that the AI already natively understands, unlocking true, zero-setup automation.

[More detailed blog post here >](#)



# The 4 questions to take to your dev or platform

If you want the technical version of “is my site agent-ready?”, these are the questions to ask.

- Do our APIs have any agentic compatibility - could they connect to something like MCP?
- Is our underlying data structured, accurate and secure enough for an agent to rely on?
- Are the third-party web apps and platforms we depend on becoming agent-ready - and what's on their roadmap?
- If we sell things: are our feeds set up for the agentic shopping channels that are emerging?



*A site that looks finished to a human can be a locked door to an agent - and you won't see it, because it still works perfectly well for you.*

## PART THREE

# The CAPTCHA paradox

The bit where the sad robot finally makes sense.

*The whole web is covered in “prove you’re not a robot.”*

*CAPTCHAs, bot-blocks, rate limits, tick-boxes.*

*All built for a good reason - keeping malicious bots out.*



# The paradox

**The exact same defences that stop BAD bots also stop GOOD agents - the ones acting for your actual customer.**

Your customer asks their AI to buy the thing or book the call. The AI reaches your site - and runs straight into a wall built to stop robots. It can't tell your security "but I'm one of the good ones." So a site can be perfectly built, fully accessible, beautifully actionable - and still slam the door at the last second.



# Nobody's fully solved this yet

There are emerging ideas - standards for agents to identify themselves, "verified agent" approaches, ways to tell a customer's agent apart from an attacker's - but it's early and contested.

The point isn't that there's an answer to buy today. It's that this tension is coming for everyone - and the businesses thinking about it now will be ahead of the ones who notice when sales start slipping.



# What you can actually do about it now

Know where your CAPTCHAs and bot-defences sit, and on **WHICH** flows.

A CAPTCHA on a login or admin page - fair enough. One sitting between a customer and your most important booking or checkout - that's now a real cost worth weighing up...

**If a form genuinely matters, ask whether its CAPTCHA is vital - because it may be turning away your customer's AI agent.**

## FORM TESTING

CAPTCHA-free critical form testing is coming soon to SiteVitals - so you're alerted if your contact form or booking stops submitting, instead of when the enquiries mysteriously dry up. Get in touch if you want more details, or check [SiteVitals.co.uk](https://SiteVitals.co.uk) if you're watching this on a replay.

## PART FOUR

# What this means - and where the opportunity is

For shops, for service businesses, and agencies.

# What changes for shops

The store that wins the agentic shift isn't the prettiest - it's the one an agent can reliably operate from search to checkout. That's a competitive advantage hiding in your codebase right now.



# What changes for service businesses

When AI starts shortlisting and contacting service providers on people's behalf, "can an agent complete our enquiry flow?" stops being a technical footnote and becomes a lead-generation question.

**The business whose form an agent can fill gets the enquiry. The one whose form it can't, doesn't - and never even knows it was in the running.**



# The opportunity for agencies

Clients can't ask AI to fix what they don't know is broken.

Knowing to test for it, and knowing how to fix it, is exactly the service that doesn't get commoditised.

**This is a genuine new service line: the agent-readiness audit.**



# SiteVitals: AI-agent accessibility, watched over time

It checks the things we've covered - semantic structure, real buttons, labelled forms, accessibility - which is now also your agent-readiness check. And on your critical CAPTCHA-free forms, it'll soon be able to test that they still submit. Because usability isn't a one-time build - it drifts. A redesign drops the semantic markup. An app update breaks a form. A new banner reintroduces layout shift. SiteVitals watches the things that break the usability you worked to build, and tells you before an agent (or a customer) hits a wall.

## FREE TO START

Uptime, an events timeline, the accessibility report and the WordPress plugin - at [SiteVitals.co.uk](https://SiteVitals.co.uk), no credit card, about 60 seconds to set up.



# The funnel is flipping

In the old funnel, a human moves through every stage - aware, interested, comparing, buying. In an agentic funnel, the AGENT does the heavy lifting across those stages, and the human mostly steps in at ONE point: to validate the choice the agent has made.

## ⚙️ THE TECHNICAL BIT

This changes what your website is FOR. It's no longer persuading a human through every step - it's giving an agent enough rich, verifiable information to confidently put YOU forward as the answer. That's the whole series in one line: May's trust, June's entity data and July's usability all feed the same job - helping an agent validate that choosing you is safe.



**Being FOUND by AI gets you considered.  
Being *USABLE* by AI gets you chosen —  
and, increasingly, actually used.**

# 5 things to test this week

Genuinely do-this-week stuff. All free, all revealing.

1. Run a [SiteVitals AI visibility report](#) - the fastest way to surface the unlabelled buttons, missing form labels and structure gaps in plain English
2. Turn JavaScript off and see what survives - if your key action vanishes, the ability for an AI to use it is greatly decreased.
3. Tab through your site with only the keyboard - if you can't reach and trigger your make-or-break action, an agent may not either
4. View source and search for `<div class="button"` - if your buttons are divs, there's your first job (or check it in the [SiteVitals](#) visibility report).
5. Find out where your CAPTCHAs sit - and whether any are blocking a form that matters.



*Scan me*

THE AGENTIC WEB SERIES

# Four sessions, one theme...

May      Why AI keeps choosing the same brands - *replay available*

---

June      The new technical SEO: what actually matters now – *replay available*

---

**July**      The agentic web: is your website usable by AI? - *today*

---

September      Building and selling websites in 2027

PART 4 OF 4 – LOOKING FORWARD TO 2027

# BUILDING AND SELLING WEBSITES IN 2027

*Date tbc: September 2026*



# Thank you!

I really hope you found this useful. Any questions - leave a comment on any post on my [Instagram](#) or [LinkedIn](#).

Grab a free [SiteVitals](#) account while you're at it – a complete set of audits on the house.

And I'd love it if you could [leave us a Google review](#) →

