websitetrouble #askwt

WEBSITETROUBLE.COM • FREE FOR WEB DESIGNERS AND SMALL AGENCIES

AI Troubleshooting Field Checklist

Client sites break at the worst times. A form stops sending. An update takes the whole site down. The client is already texting.

AI can help you fix it faster, if you know what to check first, what to ask, and how to verify the answer before it goes on a live site. This is the checklist I run before AI touches anything.

Common sense first. AI second.

20 checks	5 prompts	1 rule
In the order to run them, every time	Copy-paste, on the last pages	Never ship an AI answer you did not verify

Use it on a live job

CLIENT / SITE

WHAT BROKE, IN THE CLIENT'S WORDS

WHEN DID IT LAST WORK?

WHAT CHANGED JUST BEFORE?

Platform: ☐ WordPress ☐ Webflow ☐ Shopify ☐ Squarespace ☐ Custom / other

Who this is for: Designers and small agencies who build sites and then get the panic call when something breaks. It works with any AI tool. It does not require any paid software.

From Rob, WebsiteTrouble.com: I fix websites live, on screen share, while the owner watches. This checklist is the order I work in. Print it. Keep it next to your keyboard.

Three rules before AI touches anything

Most broken fixes I get called in to repair skipped at least one of these. AI made the skip faster, not safer.

1 Reproduce it first

See the problem yourself, on your own screen, before you change a thing. Get the exact page, the exact steps, and the exact error message or a screenshot of it. If you cannot reproduce it, you cannot know when it is fixed, and neither can AI. Half of all "mystery" bugs are one browser, one device, or one logged-in state.

2 Back it up, and know how to restore it

A backup you have never restored is a hope, not a backup. Before any fix: full site backup or host snapshot, a copy of any file or setting you are about to change, and the restore steps written down. If the host offers staging, work there first. No backup, no fix. This is the rule that keeps a bad day from becoming a lost client.

3 Change one thing, then test

One change, one test, write down what happened. AI loves to hand you five changes at once. If you apply all five and the site breaks worse, you have no idea which one did it. Slow is fast here. Keep a simple log: time, change, result.

How to ask AI so the answer is usable

Give it four things, every time:

- **The exact error** - the full message, the error code, or what the visitor sees. Not "the site is broken."
- **The context** - platform and version, theme or builder, hosting if you know it, and when it last worked.
- **What already changed** - the update, the new plugin, the edit. This is usually where the answer lives.
- **What you already tried** - so it does not send you in a circle.

Then add one line: *"Give me one step at a time, safest first, and tell me how to undo each step."*

How to verify any AI answer

- **Does that setting actually exist?** AI invents menu names and options that sound right. If you cannot find it where it says it is, stop.
- **Test the fix the way a visitor would** - logged out, on a phone, in a private window. Admin view hides most real problems.
- **Re-run the original reproduction steps.** Fixed means the exact thing that failed now works, not that something nearby looks better.
- **Check the rest of the site still works** - forms, checkout, menu, and the pages you did not touch.

Checks 1-5: Before you touch anything

Triage first. These five checks take ten minutes and stop most bad fixes before they happen.

- [ ] **1. Reproduce the problem and record it**
  Open the exact page, follow the client's exact steps, screenshot the result. Note browser, device, and whether you were logged in.

  Ask AI: *"Here is the exact error and the steps that cause it. What are the three most likely causes, in order, and how do I confirm each one?"*

  Verify: You can make it happen twice. If not, gather more detail before changing anything.

- [ ] **2. Ask what changed**
  Updates, new plugins or apps, edited settings, a new team member, a host migration. Most breakage starts within a day of a change.

  Ask AI: *"This broke after [the change]. What does that change commonly break, and what is the safest way to test if it is the cause?"*

  Verify: Check update history and recent activity logs yourself. Do not rely on memory, yours or the client's.

- [ ] **3. Back up and confirm the restore path**
  Run a full backup or host snapshot now. Save copies of files and settings you may touch. Write down how you would restore.

  Ask AI: *"On [platform / host], what is the correct way to back up this site and restore it if a fix goes wrong?"*

  Verify: The backup finished and shows today's date and a sensible file size. **No verified backup = do not proceed.**

- [ ] **4. Check whether it is just you**
  Test in a private window, on your phone on mobile data, and ask whether other visitors see it. Check the host status page and a down-for-everyone checker.

  Ask AI: *"The site fails for me but loads for others. What local causes, like cache, DNS, or a security tool, should I rule out first?"*

  Verify: You know if this is everyone, some people, or only you. Each one is a different fix.

- [ ] **5. Clear the cache before you believe anything**
  Browser cache, plugin cache, host cache, and CDN cache can all show you an old or broken version that no longer exists, or hide your fix.

  Ask AI: *"On [platform] with [cache / CDN], how do I clear every cache layer, in the right order?"*

  Verify: Reload in a private window after clearing. A surprising share of tickets end right here.

Stop rule for this section: If the site is down for everyone and the client is losing sales, say so plainly, give a time for your next update, and work the problem in order. Calm and specific beats fast and vague.

Checks 6-10: Trust and safety

The checks clients never think to run, until Google or a customer tells them something is wrong.

- [ ] **6. SSL certificate is valid and not expiring**
  Click the padlock. Check the expiry date and that every page loads on https without warnings or mixed-content errors.

  Ask AI: *"The site shows a security warning on [page]. What causes mixed content, and how do I find the exact item causing it?"*

  Verify: No warning in a private window, on desktop and phone. Note the expiry date for the client.

- [ ] **7. Domain and DNS point where you think they do**
  Confirm the domain is not expiring soon, the site loads with and without www, and recent DNS changes were intentional.

  Ask AI: *"Explain this DNS record in plain English and tell me what breaks if I change it."* Paste one record at a time.

  Verify: DNS changes can take time to spread and can take email down with the site. Change nothing here in a hurry.

- [ ] **8. Not hacked, not blacklisted**
  Run a free external scan (Sucuri SiteCheck, Google Safe Browsing). Look for pages, text, or links nobody on the team added.

  Ask AI: *"The scan reports [exact finding]. What does it mean, what do I check first, and what should I not touch?"*

  Verify: **A hack is not a normal fix.** Back up the evidence, change passwords, and escalate if you have not cleaned a hacked site before.

- [ ] **9. Updates are current, and nothing conflicts**
  List core, theme, and plugin or app versions. After a breakage, test by disabling the newest or most suspect item first, on staging where possible.

  Ask AI: *"[Item A] updated and then [symptom] started. How do I test for a conflict without taking the live site down?"*

  Verify: One item at a time, test after each. Never update everything at once on a live client site to "see if it helps."

- [ ] **10. Admin access and passwords are sane**
  Remove old users and old developer access. Confirm the client owns the domain, hosting, and site logins, not a former freelancer.

  Ask AI: *"Give me a plain-English checklist for auditing who has admin access on [platform] and what each role can do."*

  Verify: Ownership questions found during a calm audit are cheap. Found during an emergency, they are not.

Checks 11-15: Contact, forms and email

The silent failures. The site looks fine, and the business hears nothing for weeks.

- [ ] **11. Submit every form yourself**
  Contact, quote, and booking forms. Use a clear test name like "Website Test" so the client knows it is you, and confirm where the message landed.

  Ask AI: *"This form says it sent but nothing arrives. On [platform / form tool], what are the most common delivery failures, in order?"*

  Verify: **Fixed means the message arrived** in the client's inbox, not that the form showed a thank-you page.

- [ ] **12. Check spam, and email authentication**
  If mail lands in spam or never arrives, check the domain's SPF, DKIM, and DMARC records with a free lookup like MXToolbox.

  Ask AI: *"Explain this SPF / DKIM / DMARC result in plain English. What is missing, and what is the risk of changing it?"*

  Verify: One missing record is common and fixable. DNS email changes affect the whole business, so make them carefully and one at a time.

- [ ] **13. Click-to-call and contact details work on a phone**
  Tap the phone number, the email link, and the directions link on a real phone. Confirm the number matches the client's actual number everywhere it appears.

  Ask AI: *"The phone link does nothing on mobile. What markup makes a number tap-to-call, and how do I test it?"*

  Verify: On a phone, not in a desktop browser resized small. The two are not the same test.

- [ ] **14. Booking and checkout links are alive**
  Follow every "book now," "buy," and calendar link to the end. Dead booking links and expired scheduling pages lose the sale quietly.

  Ask AI: *"Visitors say the booking page errors after choosing a time. What should I check on [booking tool] before changing the website?"*

  Verify: You reached a real, current booking or checkout page. Do not complete a fake purchase unless the client asked you to.

- [ ] **15. Findable: Google can see the site**
  Search the business name and "site:" plus the domain. Check the Google listing shows the right phone, hours, and website link.

  Ask AI: *"The site does not appear when I search [exact search]. What are the likely technical reasons, and how do I rule each out?"*

  Verify: A page blocked from indexing, a wrong listing link, or a duplicate site is a finding. Write it down even if today's job was something else.

Checks 16-20: Fast and working

What visitors actually experience, especially on a phone, on a slow connection, away from the office.

- [ ] **16. Load speed on a phone**
  Run PageSpeed Insights and read the mobile score first. Look for the one or two biggest items it names, not all forty.

  Ask AI: *"PageSpeed flags [exact item]. Explain what it means in plain English and the safest fix on [platform]."*

  Verify: Re-run the test after your change. A faster score that broke the layout is not a fix.

- [ ] **17. No broken links or missing pages**
  Run a free broken-link checker. Fix or redirect the ones visitors and Google actually use, starting with menu, footer, and top pages.

  Ask AI: *"These pages return 404. Which should redirect, which should be rebuilt, and what makes a good redirect target?"*

  Verify: Click the fixed links yourself. A redirect to the homepage for everything is a shortcut visitors notice.

- [ ] **18. Images load and are not crushing the page**
  Look for broken image boxes and multi-megabyte photos straight from a camera. Compress and resize before re-uploading.

  Ask AI: *"This image is [size] and displays at [dimensions]. What size and format should it be, and what will I lose by compressing it?"*

  Verify: The page looks the same and loads faster. Keep the originals in your backup.

- [ ] **19. The whole site works on a real phone**
  Menu opens, text is readable without zooming, buttons are tappable, nothing runs off the side of the screen. Test the top five pages.

  Ask AI: *"On mobile, [element] overflows / will not open on [builder or theme]. What are the usual causes and the safest test fix?"*

  Verify: Most client traffic is mobile. A site that only works on your desktop does not work.

- [ ] **20. The build did not break the basics**
  After any AI-assisted build or fix, walk the site like a new visitor: home, services, about, contact, one conversion action, start to finish.

  Ask AI: *"I just changed [what you changed]. Give me a short regression checklist of what this kind of change commonly breaks."*

  Verify: Then log what you changed, when, and the backup you can restore. Future you, or the next designer, will need it.

The pattern to notice: None of these 20 checks required AI. AI gets useful after them, when you hand it a real error, real context, and one clear question. That order is the whole system.

Five prompts worth keeping

Copy, paste, fill in the brackets. Each one asks for one step at a time on purpose.

1. The diagnosis prompt
"My client's site on [platform] shows this exact error: [paste error]. It started after [change]. Give me the three most likely causes in order, and a safe test for each one. One step at a time."

2. The plain-English prompt
"Explain [report, record, or log line] in plain English. Tell me what breaks if I change it, and what I should check before I touch it."

3. The safe-fix prompt
"Give me the safest fix for [problem] on [platform], step by step. For each step, tell me how to undo it. Flag anything that could take the site or email down."

4. The code-check prompt
"Here is code you suggested earlier / code I found: [paste]. Explain what every part does, what it could break, and where exactly it goes. If you are unsure, say so."

5. The client-update prompt
"Turn these fix notes into a short, calm client update: what happened, what I did, what I tested, what happens next. No jargon. Notes: [paste]."

Red flags: when AI is confidently wrong

- It names a menu, setting, or button you cannot find. It may not exist on your platform.
- It gives five changes at once, with no way to tell which one worked.
- It says edit core files, the database, or DNS, with no backup and no undo.
- Its answer changes when you ask twice. Verify harder, not faster.
- It sounds certain about your client's host or account. It cannot see them. You can.

Charge for this work

Troubleshooting is a service, not a free favor that comes with a design project. Name it, scope it, and quote it before you start.

Stuck right now? If a client site is down and this checklist is not enough, that is my actual day job. At WebsiteTrouble I fix broken sites live, on screen share, while you watch, and I explain what I am doing as I go. No ticket queue, no waiting on email. Get help at websitetrouble.com.

Fix It With AI is from WebsiteTrouble.com, live website help from a real human, on screen share, while you watch. This checklist teaches a working method. It does not promise clients, income, or results.