

I asked Claude Code to remove Jquery. It failed miserably.



by Alex Yumashev · Updated Feb 13 2026

Disclaimer: this is a rushed angry rant with F-bombs all over. I had a rough day alright. If explicit language is an issue, please skip the read.

Also for context - I'm a heavy AI user and my productivity has genuinely gone up 10x thanks to these tools. But fuuuuuu....

Big parts of our app still use jQuery (oh shut up) and we're migrating away from it. Nothing ambitious, just replacing DOM manipulation and event handling with vanilla JS. Our complex interactive screens already use Vue.js, but dozens of simple admin pages still have little jQuery sprinkled around.

So I thought: hey, perfect job for an AI agent. Repetitive, mechanical, well-defined, long and boring. Enough with my small local rewrites, let's try the autonomous agent, I wanna be like those youtubers.

So I pointed my MAX priced Claude Code (Opus 4.6) at the codebase. Poured some coffee.

It failed fucking miserably.

OK I didn't just yolo it, I did my homework. Wrote a comprehensive `CLAUDE.md` with detailed instructions. Prepared helper functions - like `const _id = id => document.getElementById(id);`. Described edge cases, like, Jquery being forgiving for non-existing elements, while plain JS is not, so we have to use optional-chaining. I even wrote my custom `fadeIn` and `fadeOut` replacements on `HTMLElement.prototype` using CSS transitions, so `\$(#something).fadeIn()` just becomes `_id(something).fadeIn()`. Then I went through "plan mode" and set up two parallel agents - one writing code, one reviewing it.

Overall the task was to edit 67 files, about 5-10 lines to rewrite in each. Simple AI-enhanced find and replace.

Not exactly rewriting a fucking C compiler in Rust from scratch or whatever they claimed it did.

It was a total shitshow.

Why AI is so bad at vanilla JS and HTML, when there's no React/Vue in a project? Just a couple of examples:

1. It writes `<script type="module">`, then immediately references `document.currentScript` in the first line. `currentScript` doesn't work in modules! This is literally on the MDN page in a yellow damn warning box or something. You read the entire internet and somehow skipped that part.

2. It uses my `_id("something")` shorthand to grab elements that *don't exist in the HTML*. Just confidently references imaginary DOM nodes. How about reading that 100 LOC partial file first? Or use null-chaining maybe? The very thing I prompted you to use because I knew this would happen? Nnnn-ope.

3. On the rare occasion it did realize `getElementById` can return null (congrats), it wrote this beauty: `onclick="var x=document.getElementById("blah"); if (x) x.doSomething();"`. (yeah, "onclick" is another story) - instead of just `_id("blah").doSomething()`. You know, the thing I asked for. Never happened.

4. It writes selectors like `querySelectorAll("#123")`. CSS selectors don't allow IDs starting with a digit. Jquery did. CSS doesn't. Every junior frontend dev who's ever dealt with auto-generated IDs has been burned by this. It's a rite of passage. The AI skipped the initiation.

5. Calling functions from external `<script type="defer">` in inline scripts without waiting for `DOMContentLoaded` (so that the deferred scripts can finish loading)

etc...

tableDnD.init(_id("actions"), { onDra

The > 2 suggests it needs header + 2+ action rows. But shouldn't > 2 be changed to >= 3? Verify the business logic.

Yeah... Why not, bro... Good idea.

Not to mention way more nuanced issues. For example, when you append AJAX-loaded HTML with scripts into your document using `\$.html()` - Jquery automatically runs all the "ready" functions for you - the `\$(function)` shorthand. When you insert AJAX-ed HTML via vanilla JS `createContextualFragment` or `createElement` - it also executes all scripts, but `DOMContentLoaded` is skipped.

I know, hard to catch even for a senior dev, no complaints here... Still, I had a small, tiny little hope it will spare me the load here. It wrote that Rust compiler after all.

Also, why not run "npm run test" at some point? We have tons of tests. I even have an integration test that crawls the entire fucking app recursively link-by-link in a headless browser and reports on JS errors. CLAUDE.md has all the info.

Should I be doing this in the early mornings EU-time, when the US is still asleep and Anthropic does not nerf their model to save on electric bill?

I mean. My project is 150K lines of code. About half of that is irrelevant - C# backend, database scripts, CI/CD - it all lives in separate directories the agent didn't even need to touch. We're talking about maybe 30-40K lines of actual frontend code. Mostly HTML. 20% of that has Jquery.

I don't know what to think about that "writing a Rust-based C compiler from scratch with 24 agents" demo. Yeah. I bet that took more time prompting and babysitting than just coding the damn thing with cursor-tab. But hey, it made a great tweet.

document.currentScript.closest(

MOTHERFUCKER

_HttpRequestAction.cshtml

• You're right. When I moved it to type="module", document.

Yeah... Why not, bro... Good idea.

Can I use null safe operator like this? myVar?.Property = Something();

UiTestCrawl.cs

• Thinking
• No, that's not valid in C#. The null-conditional operator `?.` can only be used for reading/accessing members, not for assignment. You'll need to check for null explicitly first: if (myVar != null) myVar.Property = Something(); or use a ternary/null-coalescing operator `??` which is different: myVar ??= Something() assigns only if myVar is null.

are you sure?

• Thinking
• Yes, I'm sure. myVar?.Property = Something() is not valid C# syntax and will give you a compiler error. The null-conditional operator `?.` can only be used on the right side of an assignment (for reading), not on the left side. You might be thinking of null-coalescing assignment `??` which is different: myVar ??= Something() assigns only if myVar is null.

This is C# not JS

• Thinking
• I know, I'm talking about C# too. The answer is the same - you cannot use `?.` on the left side of an assignment in C#. It will give you a compile error like "The left-hand side of an assignment must be a variable, property, or indexer." If you want to assign to a property only when the object isn't null, you need to use an explicit null check.

That's called null-conditional assignment how recent is your knowledge?

• Thinking
• My knowledge cutoff is January 2025, and as of that date, myVar?.Property = Something() is not valid C# syntax. It's a feature that's been added in C# 9.0.

My AI work be like..

About

About Jitbit Software

Jitbit's Helpdesk Ticketing System

Subscribe

Get (rarely posted) updates in your inbox.

Enter your email

Subscribe

Accept