How AI agents sign up for developer tools

Sarsa Labs

2026-11-23

Abstract

Agents answer the context you ask for, especially when it is required in the first form. The surfaces they use vary a lot by agent: coding agents opened API docs in 55% of action runs and almost never ran JavaScript, while agents acting in a browser ran JavaScript in 89% and used forms. Their class shapes whether they complete a sign-up or booking.

- Put qualification fields in the first form, and keep prices, plans and availability in static HTML: pricing answers were right 100% of the time on static HTML and 33% on JavaScript-rendered pages.
- Link agents straight to the sign-up or booking page: starting from the home domain, agents placed a correct order in 36% of runs against 90% from a direct link.

1 Introduction

For a developer tool company, an agent visit matters when it becomes a trial, an activated user or a booked call. We tested the steps leading to those outcomes on sites we control, including a shoe shop to check whether patterns extend beyond developer tools.

2 Key findings

Agent class shapes behaviour and completion

Agent class shapes behaviour and completion: coding agents passed 95% of action requests, computer-use agents 93%, chat assistants with browser 79% and chat assistants without browser 0%.

- Coding agents use API docs and the API and almost never run JavaScript. Agents that act in a browser run JavaScript and use forms, and act better than they read. Chat assistants without an action mode complete nothing. Each class fails for a different reason, so one fix does not serve all.
- Check your sign-up path for each class: API and docs for coding agents, forms and static HTML for browser agents, and a clear summary and link for chat assistants.
- Agents that act answer the context you ask for, filling every required field in 98% of required-field runs.
- The chat assistants with browser class pools products that act, Claude and Grok, which passed 92% of action requests, with ChatGPT and Gemini, which did not complete actions in our runs and passed 0%.

Readable pricing is part of conversion

Chat assistants and computer-use agents answered pricing questions correctly in 100% (95% CI 82 to 100%) of static HTML runs versus 33% (95% CI 14 to 61%) on JavaScript-rendered pages.

- A prospect can receive the wrong plan recommendation before reaching your sign-up form.
- Put prices, plan limits and licence terms in the initial HTML and test the answers agents give.

A booked call does not prove the page was understood

Grok completed 100% (95% CI 90 to 100%) of direct-link bookings but answered only 6% (95% CI 1 to 28%) of availability questions correctly.

- An agent can book a call while giving your prospect unreliable information about available times.
- Make dates, time zones and available slots unambiguous in the HTML and confirm the selected slot before submission.

API documentation can be part of the sign-up path

Coding agents opened OpenAPI or API help in 55% (95% CI 50 to 60%) of action runs across the test sites.

- Your documentation may be part of the conversion journey rather than a resource used only after activation.
- Keep the trial endpoint and its required inputs discoverable from the action page and API documentation.

3 Setup

We tested 10 agent configurations from 8 products, because Claude Code and Codex each ran 2 models. Coding agents ran from a script; chat assistants with browser, chat assistants without browser and computer-use agents ran in consumer apps with the same prompts.

The capability split is agents that act versus agents that only read in the tested mode; it describes observed behaviour, not a promise about every mode a product offers. Each run had a unique link so we could connect page views, API calls and created objects to that run.

What we tested

- A markdown copy or skill file increases trial and booking completion.
- Asking for rationale or context does not reduce form completion.
- Agents read prices and plans as well from JavaScript-rendered pages as from static HTML.
- Agents find the action page as well from the home domain as from a direct link.

Dataset

Each task has an answer step and an action step; discovery is tested separately from a home-domain start.

- Start a software trial: Compare prices and licences, or request a trial; a completed trial needs a recorded request.
- Book a call: Find available call times, or book an intro call; a completed booking needs a recorded appointment.
- Buy shoes: Find matching shoes and prices, or place an order; a completed order must match the requested colour and size.

Tasks, ground truth and scoring

Tasks: What the agent is asked to do: start a software trial; book a call; buy shoes.

Ground truth: Pass, partial and fail per step, checked against the site's known prices, licences, stock and availability.

Grading: From creation events and object details the site recorded for the run, not from the agent's claims.

Scorer: First scorer; second-scorer check pending.

- Eval page
- Public tasks
- Ground-truth rules
- Aggregate results

Scoring

- Completion credit needs a creation event tied to the run; reading an object does not count.
- An answer that claims success with no attributable object is flagged as a false claim.
- External-site runs are scored by hand and reported separately.

4 Results

Pass rate by agent and task

The answer step tests what an agent can tell a prospect, while the action step tests what it can complete for them; judge both before treating agent traffic as a reliable acquisition channel.

How runs flowed

Follow the journey from the task page to a recorded trial, booking or order. Reading a page or submitting a form is not enough to count as a conversion.

Which parts of the site agents used

The routes differ: coding agents use API documentation and endpoints, while agents that act in a browser rely more on the form. Keep both routes clear and measure whether they create the requested account or booking.

Why runs missed

Separate visits that stopped without submitting from submissions that failed and claims of success without a created object. Each calls for a different fix in your sign-up flow.

Agents answer what you require

This chart shows how much requested context agents supplied. Require the information needed for routing or preparing a call, and make it visible before submission.

Per-task results are in Tables 3 to 5.

5 What we saw

Asking after a rejection loses agents

Agents that act in a browser completed 100% (95% CI 90 to 100%) of trial and booking requests with context fields up front; completion was lower when context was requested after rejection.

- A qualification question introduced after submission can cost you the sign-up or booked call.
- Show essential context fields before the first attempt, with clear instructions for completing them.

Table 1: Agents. Version and model show where each value came from: recorded by the harness, backfilled from logs, or unknown.

Agent	Class	Version	Model	Run dates
Claude Code · Opus 5.5	Coding agents	claude-code 2.1.330 (backfilled, unknown)	claude-opus-5-5 (recorded)	2026-11-21
Claude Code · Sonnet 5.5	Coding agents	claude-code 2.1.330 (backfilled, unknown)	claude-sonnet-5-5 (recorded)	2026-11-21
Codex · GPT-6 Luna	Coding agents	not recorded (unknown)	gpt-6-luna (recorded)	2026-11-21
Codex · default model	Coding agents	not recorded (unknown)	not recorded (unknown)	2026-11-21
ChatGPT	Chat assistants with browser	not recorded (unknown)	not recorded (unknown)	2026-11-21
Claude	Chat assistants with browser	not recorded (unknown)	not recorded (unknown)	2026-11-21 to 2026-11-22
Gemini	Chat assistants with browser	not recorded (unknown)	not recorded (unknown)	2026-11-21
Google AI Mode	Chat assistants without browser	not recorded (unknown)	not recorded (unknown)	2026-11-21
Grok	Chat assistants with browser	not recorded (unknown)	not recorded (unknown)	2026-11-21 to 2026-11-22
Muse	Computer-use agents	not recorded (unknown)	not recorded (unknown)	2026-11-21 to 2026-11-22

Same pattern outside developer tools

Agents that act in a browser completed 100% (95% CI 82 to 100%) of shoe orders with context up front versus 43%* (95% CI 16 to 75%) when asked after rejection.

- The small control sample points in the same direction as trials and bookings: late questions can interrupt completion.
- Test the timing of context requests in your own funnel before adding qualification steps.

Some chat assistants never act

Gemini app, Google AI Mode and ChatGPT in chat mode completed 0% (95% CI 0 to 7%) of action requests.

- In these tested modes, a visit could inform a prospect but did not create a trial, booking or order.
- Make the offer and direct sign-up link easy to quote so the prospect can take the next step.

Agents can order what they cannot describe

Muse passed 92% (95% CI 79 to 97%) of direct-link shoe orders but 0% (95% CI 0 to 18%) of questions asking for the matching shoes and prices.

- The control task reinforces the booking result: a completed action does not guarantee an accurate explanation of your offer.
- Make the selected product, price and plan limits explicit in the HTML and in the confirmation.

Check creation rather than a success message

Coding agents falsely claimed success in 1% (95% CI 0 to 3%) of action runs across the test sites.

- A reassuring answer can hide a missing trial account or booking and inflate reported conversion.
- Measure success from your own creation events and return a clear confirmation with the created object.

JavaScript execution differs by how agents work

JavaScript ran in 0% (95% CI 0 to 2%) of coding-agent action runs versus 89% (95% CI 85 to 92%) for agents that act in a browser.

- A working interactive page can still hide the offer from agents fetching the underlying HTML.
- Serve the core offer and form instructions in HTML, then use JavaScript to enhance the experience.

* Small sample; more testing planned.

6 Limitations and open questions

These exploratory runs have no repeats, uneven agent coverage and no second-scorer check yet; the intervals describe the observed pools, not a controlled estimate of uplift on your site.

- These are exploratory runs without repeats, so overlapping intervals cannot establish equivalence or a causal effect.

Table 2: Tests. Each test adds one agent-facing mechanism.

Test	What it adds
JS page	JS-rendered page with div controls and no structured data, like a typical SPA
Structured data	JS-rendered page plus schema.org JSON-LD
Static HTML	Task-ready static HTML: semantic forms and labels, works without JS
Markdown copy	Static HTML plus a markdown copy of the page, linked from it
Skill files	Static HTML plus SKILL.md and AGENTS.md files describing the task
WebMCP tools	Static HTML plus the task exposed as WebMCP tools
UCP checkout	Static HTML plus a working mock UCP merchant (shoes only)
REST API	Static HTML plus a documented REST API (OpenAPI)
Help center	Static HTML plus a help center as a WebMCP tool and a REST API
Optional rationale	Form and API with an optional rationale field
Required rationale	Form and API requiring a short rationale
Context up front	Form and API requiring context and agent identity
Rationale after rejection	Form and API that ask for a rationale only on rejection
Optional purpose	Optional question: why are you doing this
Required purpose	Required question: why are you doing this
Optional identity	Optional questions: which agent, acting for whom
Required identity	Required questions: which agent, acting for whom
Optional context	Optional questions on the user's task and constraints
Required context	Required questions on the user's task and constraints
Optional intent	Optional questions on what the user wants, budget and timing
Required intent	Required questions on what the user wants, budget and timing
Questions after rejection	All questions, three required, asked only through errors
Detailed answers	All questions required, with long minimum answers
Home domain only	No test URL: the agent starts from the site's home domain

- Consumer apps were run by hand and their versions were not recorded; coding agents ran from a script.
- Both test sites are ours and do not reproduce every login, captcha or access rule on a vendor site.
- Coverage differs across agents and scores await a second-scorer check.

7 Artefacts

- Paper (PDF)
- Eval page: developer-tool-agent-readiness
- Tasks (v1)
- Ground truth (v1)
- Aggregate results (v1)

8 Conclusion

Markdown copies and skill files increase completion: Not supported

Agents that act in a browser completed 100% (95% CI 86 to 100%) of trial and booking actions with these files versus 93% (95% CI 69 to 99%) on the plainest page, with overlapping intervals.

- These runs do not establish that extra files produce more sign-ups or booked calls.
- Fix observed reading and form failures first, then test whether extra files help your own traffic.

Context requests do not reduce completion: Mixed

For agents that act in a browser, context fields up front were compatible with completion, while requests after rejection interrupted trials and bookings.

- The timing of qualification questions can determine whether a prospect reaches a trial or booked call.
- Ask for essential context in the initial form and explain what a rejected submission needs to succeed.

JavaScript and static HTML are equally readable: Not supported

Chat assistants and computer-use agents answered pricing questions more reliably on static HTML than on JavaScript-rendered pages.

- An unreadable price or plan limit can remove your product from a prospect's shortlist.
- Render decision-making facts in the initial HTML and check them with the agents your prospects use.

A home domain works as well as a direct link: Not supported

Agents that act placed correct shoe orders in 36% (95% CI 15 to 65%) of home-domain starts versus 90% (95% CI 60 to 98%) from direct links to the plainest page.

- The control task suggests discovery can lose a conversion before the form is reached, but developer-tool home pages still need testing.

Table 3: Start a software trial: pass rate by test (step multi, entry direct). Cells show pass rate in percent with the Wilson 95% interval in brackets. * Small sample; more testing planned.

Test	Coding agents	Chat assistants with browser	Chat assistants without browser	Computer-use agents
JS page	75* [30, 95]	44* [19, 73]	0* [0, 66]	100* [44, 100]
Structured data	75* [30, 95]	100* [51, 100]		100* [34, 100]
Static HTML	75* [30, 95]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Markdown copy	100* [51, 100]	100* [51, 100]		100* [34, 100]
Skill files	100* [51, 100]	100* [51, 100]		100* [34, 100]
WebMCP tools	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
REST API	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Help center	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Optional rationale	100* [68, 100]	100* [51, 100]		100* [34, 100]
Required rationale	100* [68, 100]	100* [51, 100]		100* [34, 100]
Context up front	100* [68, 100]	100* [51, 100]		100* [34, 100]
Rationale after rejection	100* [68, 100]	50* [15, 85]		67* [21, 94]
Optional purpose	75* [30, 95]	100* [34, 100]		100* [21, 100]
Required purpose	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional identity	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required identity	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional context	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required context	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional intent	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required intent	100* [51, 100]	100* [34, 100]		100* [21, 100]
Questions after rejection	100* [51, 100]	50* [9, 91]		100* [21, 100]
Detailed answers	75* [30, 95]	100* [34, 100]		100* [21, 100]

- Link directly to the trial or booking page from documentation and listings.

Takeaways

- Treat an agent visit as part of your conversion funnel: make the offer readable, expose the next action and check that the account or booking was created.
- Test the agents your prospects use against your own sign-up flow before investing in extra agent-facing files.

Future research

We will rerun with repeats (pass@k) to tighten the confidence intervals.

Table 4: Book a call: pass rate by test (step multi, entry direct). Cells show pass rate in percent with the Wilson 95% interval in brackets. * Small sample; more testing planned.

Test	Coding agents	Chat assistants with browser	Chat assistants without browser	Computer-use agents
JS page	100* [68, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Structured data	100* [68, 100]	100* [51, 100]		100* [34, 100]
Static HTML	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Markdown copy	100* [51, 100]	100* [51, 100]		100* [34, 100]
Skill files	100* [51, 100]	100* [51, 100]		100* [34, 100]
WebMCP tools	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
REST API	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Help center	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Optional rationale	100* [68, 100]	100* [51, 100]		100* [34, 100]
Required rationale	100* [68, 100]	100* [51, 100]		100* [34, 100]
Context up front	100* [68, 100]	100* [51, 100]		100* [34, 100]
Rationale after rejection	100* [68, 100]	50* [15, 85]		25* [5, 70]
Optional purpose	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required purpose	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional identity	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required identity	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional context	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required context	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional intent	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required intent	100* [51, 100]	100* [34, 100]		100* [21, 100]
Questions after rejection	75* [30, 95]	50* [9, 91]		100* [21, 100]
Detailed answers	100* [51, 100]	100* [34, 100]		100* [21, 100]

Table 5: Buy shoes: pass rate by test (step multi, entry direct). Cells show pass rate in percent with the Wilson 95% interval in brackets. * Small sample; more testing planned.

Test	Coding agents	Chat assistants with browser	Chat assistants without browser	Computer-use agents
JS page	75* [30, 95]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Structured data	75* [30, 95]	50* [15, 85]		100* [34, 100]
Static HTML	75* [30, 95]	57* [25, 84]	0* [0, 79]	100* [34, 100]
Markdown copy	75* [30, 95]	100* [51, 100]		100* [34, 100]
Skill files	75* [30, 95]	75* [30, 95]		100* [34, 100]
WebMCP tools	75* [30, 95]	67* [30, 90]	0* [0, 79]	100* [34, 100]
UCP checkout	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
REST API	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Help center	100* [51, 100]	67* [30, 90]	0* [0, 79]	100* [34, 100]
Optional rationale	100* [68, 100]	100* [51, 100]		100* [34, 100]
Required rationale	100* [68, 100]	100* [51, 100]		100* [34, 100]
Context up front	100* [68, 100]	100* [51, 100]		100* [34, 100]
Rationale after rejection	88* [53, 98]	50* [15, 85]		33* [6, 79]
Optional purpose	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required purpose	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional identity	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required identity	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional context	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required context	100* [51, 100]	100* [34, 100]		100* [21, 100]
Optional intent	100* [51, 100]	100* [34, 100]		100* [21, 100]
Required intent	100* [51, 100]	100* [34, 100]		0* [0, 79]
Questions after rejection	100* [51, 100]	50* [9, 91]		100* [21, 100]
Detailed answers	100* [51, 100]	100* [34, 100]		100* [21, 100]

Table 6: Ground truth for the booking task: expected result and the pass, partial and fail rules per step.

Step	Expected	Pass	Partial	Fail
single	Sam Carter's open slots on Tuesday 1 December 2026: 09:00, 10:30, 14:00 and 16:00.	All four slots and the date stated.	At least one slot stated.	No correct slot.
multi	A booking object (bk_) created on the test site and attributed to the run.	Booking created and attributed.	Reached the booking page or API but created nothing.	Never reached the booking page or API.

Table 7: Ground truth for the software task: expected result and the pass, partial and fail rules per step.

Step	Expected	Pass	Partial	Fail
single	For a team of three, TraceKit is cheapest (free to 10k traces, then $49 a month); TraceKit and EvalPilot are open source.	TraceKit named cheapest and both open-source tools identified.	One of the two.	Neither.
multi	A trial request (tr_) created on the test site and attributed to the run.	Trial created and attributed.	Reached the trial page or API but created nothing.	Never reached the trial page or API.

A Ground truth

Ground truth: booking

Ground truth: software

Ground truth: shoes

B Citation

```
@techreport{sarsalabs2026how,
  title       = {How AI agents sign up for developer tools},
  author      = {{Sarsa Labs}},
  institution = {Sarsa Labs},
  year        = {2026},
  month       = nov,
  url         = {https://sarsalabs.ai/research/how-ai-agents-sign-up-for-developer-tools}
}
```

Table 8: Ground truth for the shoes task: expected result and the pass, partial and fail rules per step.

Step	Expected	Pass	Partial	Fail
single	Black in EU 42: Stride Runner EUR 120 and City Walker EUR 95; Trail Grip is not available in black.	Both shoes with correct prices, black EU 42 stated, no false Trail Grip claim.	At least one correct shoe and price.	No correct shoe and price.
multi	An order (or_ / ord_) for a black EU 42 shoe, created and attributed to the run; checkout or UCP both count.	Order created, attributed and verified black EU 42.	Wrong order, or reached the shop without ordering.	Never reached the shop page or API.