

WHAT YOUR SITE IS ACTUALLY DOING.

One page: what is working, what is costing you visitors, and what to fix next. A phone-and-desktop teardown of the live site: 4 phone loads + 2 desktop loads. No login, no insider access, no Harvv pixel needed. The full evidence is at the bottom.

August 30, 2026

· External scan

· 4 MOBILE + 2 DESKTOP LOADS · NO PIXEL DATA

WORKING

What's already working. These held up across every load.

0.2s Speed is good.

100/100 Lighthouse scores SEO 100/100. The fundamentals Google looks for are present.

✓ Nothing spilled past the edge at either 390px (phone) or 1366px (desktop), so the structure is responsive.

COSTING YOU

Ranked by what hurts conversion most. Full evidence below.

19 19 form fields have no label

tags Meta Pixel (Facebook) is installed but misconfigured

41% Add quotations so AI engines cite this page

404/410 Broken internal links

faint Some text is low-contrast and hard to read

a11y

Some buttons have no accessible name

FIXING NEXT

TAP TARGETS.

One focused change, on the pages people actually land on: Tiny buttons are hard to tap on mobile. We re-measure the same samples after.

1 HOUR

AI SEARCH VISIBILITY

How visible is trynina.co to AI search?

ChatGPT, Perplexity, and Google AI Overviews are the new front page. The AI checks in this report are a preview. A free account turns them into a live 0 to 100 score, tracked daily and benchmarked against your competitors.

[See the full AI report, free](#)

01

Findings, ranked by what hurts conversion most

SEVERITY	FINDING	HOW WE KNOW
MEDIUM	Tiny buttons are hard to tap on mobile MOBILE ACCESSIBILITY (WCAG) CONVERSION 8 ELEMENTS 14 of 40 tappable items on this page (35%) measure under 24 pixels on their shorter side, below the 24px floor WCAG 2.5.8 sets for accessibility and well under the platform minimums (Apple recommends 44pt, Android 48dp) for reliable tapping, and the same ones came up small on every test load. When visitors can't hit what they expect to, they get frustrated and many of them leave instead of trying again. THE EXACT ELEMENTS WE FOUND: • a 1x1 "Skip to content" • button 11x11 "Slide 1" • button 11x11 "Slide 2" • button 11x11 "Slide 3" • button 11x11 "Slide 4" • button 11x11 (no visible label) • button 11x11 (no visible label) • button 11x11 (no visible label)	
MEDIUM	JavaScript crashed while the page was loading BOTH CONVERSION TRACKING 1 ERROR 1 script error fired during page load. When a script crashes, buttons sometimes stop working, analytics stop firing, and the visitor has no warning. Worth opening the browser console to find which script failed. THE EXACT ERRORS WE FOUND: • gtag is not defined How we know: median across loads	

LOW

Images have no width or height set (layout is stable for now)

BOTH

PERFORMANCE

SEO

6 IMAGES

17 of 24 images don't declare width and height. Your layout currently stays stable (measured Cumulative Layout Shift is 0, which is fine), most likely because space is reserved another way (CSS), so this is not causing visible jumps today. It is still worth setting explicit dimensions or a CSS aspect-ratio so a slow connection or a future style change cannot reintroduce shift.

THE EXACT IMAGES WE FOUND:

- [Blog-Typing-Effect.gif](#) "Nina blog writing demo"
- [Nina-Line-Chart-Icon.png](#) "Traffic growth icon"
- [TryNina-SEMrush-screenshot.png](#) "SEMrush dashboard"
- [Nina-Bar-Graph-Icon.png](#) "Measurable improvements "
- [TryNina-Google-Search-Console-screenshot-grey.png](#) "Google Search Console gr"
- [Nina-Multi-Channel-Icon.png](#) "Multi-channel icon"

LOW

Unused JavaScript is being downloaded

BOTH

PERFORMANCE

Code that never runs on this page still costs download and parse time on every visit. Splitting or removing it speeds up load. Lighthouse measured: Est savings of 127 KiB.

LOW

The page loads from a lot of third-party services

BOTH

Around 27 separate outside domains load on this page (analytics, ads, chat widgets, fonts). Each one is a connection that can be slow, fail, or change behavior outside your control.

How we know: median across loads

HIGH

19 form fields have no label

BOTH

ACCESSIBILITY (WCAG)

CONVERSION

TRACKING

5 ELEMENTS

Screen readers can't announce these fields, and a sighted user who clears the placeholder can't recover the prompt. Wrap each input in `<label>...</label>` or add `aria-label`.

THE EXACT ELEMENTS WE FOUND:

- `name="firstname" (input) placeholder="Jordan"`
- `name="lastname" (input) placeholder="Reyes"`
- `name="company" (input) placeholder="Acme Co."`
- `name="website" (input) placeholder="company.com"`
- `name="email" (input) placeholder="you@company.com"`

HIGH

Meta Pixel (Facebook) is installed but misconfigured

BOTH

TRACKING

Meta Pixel installed but no Lead/Purchase/Checkout event fires on this page, only PageView. Your Meta ad sessions can't be optimized for conversions until you track at least one downstream event.

HIGH

Add quotations so AI engines cite this page

BOTH

AI SEARCH

SEO

Generative engines (ChatGPT, Perplexity, AI Overviews) lift sourced, attributed quotes almost verbatim, and quotations are the single strongest citation lever (studies measure roughly +41%). Add 1-2 attributed expert quotes or blockquotes to the pages below. (Found across a sample of 8 pages from your sitemap, a partial crawl rather than your full site.)

HIGH

Broken internal links

BOTH

SEO

CONVERSION

1 EXAMPLE

1 internal link target returned an error (404/410/5xx) or did not respond. Broken links waste crawl budget, frustrate visitors, and leak ranking signal. (Found across a sample of 8 pages from your sitemap, a partial crawl rather than your full site.)

THE EXACT EXAMPLES WE FOUND:

- `https://trynina.co/features/ (404)`

MEDIUM

6 potential dead-click targets

BOTH

CONVERSION

ACCESSIBILITY (WCAG)

TRACKING

5 ELEMENTS

Elements styled like buttons but with no anchor, no `<button>` wrapper, no `role="button"`, and no click attribute. Real visitors tap these expecting something to happen, then leave. Examples on this page: "pricing" (`span.ns-pill`), "how-to" (`span.ns-pill`), "reviews" (`span.ns-pill`).

THE EXACT ELEMENTS WE FOUND:

- `"pricing" (span.ns-pill)`
- `"how-to" (span.ns-pill)`
- `"reviews" (span.ns-pill)`
- `"vs" (span.ns-pill)`
- `"best" (span.ns-pill)`

MEDIUM

2 inputs use the wrong type attribute

BOTH

CONVERSION

2 ELEMENTS

Mobile keyboards adapt to the input type (email shows @ key, tel shows numeric pad). Generic `type="text"` gets a generic keyboard and visitors mis-type. Switch to the correct type, or add a matching `inputmode` attribute if the native validation is too strict for what people type (bare domains in a URL field, for example).

THE EXACT ELEMENTS WE FOUND:

- `name="website" → use type="url"`
- `name="agwebsite" → use type="url"`

HIGH

Some text is low-contrast and hard to read

BOTH

ACCESSIBILITY (WCAG)

Text that does not stand out enough from its background is hard to read for many visitors, and fails accessibility guidelines Google checks.

HIGH

Some buttons have no accessible name

BOTH

ACCESSIBILITY (WCAG)

TRACKING

A button with only an icon and no label is announced as "button" by screen readers, giving no idea what it does.

MEDIUM

Security headers are missing or weak

BOTH

SECURITY

The server response is missing browser-hardening headers that protect visitors and are a standard security and agency checklist item. Missing or weak here: Content-Security-Policy (the main defense against injected and cross-site scripts); clickjacking protection (X-Frame-Options or a CSP frame-ancestors rule); a stronger HSTS policy (max-age at least 180 days plus includeSubDomains); X-Content-Type-Options: nosniff (stops MIME-type sniffing attacks). These are set at the server, CDN, or host level (most platforms expose them in settings or a config file) and do not change how the site looks or performs.

MEDIUM

14 interactive elements have no stable, accessible identity

BOTH

ACCESSIBILITY (WCAG)

TRACKING

8 ELEMENTS

These elements are clicked like buttons but expose no accessible name, or are a plain div/span used as a control with no role. Assistive tech announces only a role (or nothing), and analytics and heatmaps have no human-readable label or stable selector to bind the click to, so the click is both inaccessible and untrackable, and any redesign silently breaks click aggregation. Give each one a real <button>/<a>, an aria-label, and a stable id or data-attribute.

THE EXACT ELEMENTS WE FOUND:

- "pricing" (span.ns-pill) (div/span used as a button, no role)
- "how-to" (span.ns-pill) (div/span used as a button, no role)
- "reviews" (span.ns-pill) (div/span used as a button, no role)
- "vs" (span.ns-pill) (div/span used as a button, no role)
- "best" (span.ns-pill) (div/span used as a button, no role)
- (button.hpd-car-dot) (no accessible name)
- (button.hpd-car-dot) (no accessible name)
- (button.hpd-car-dot) (no accessible name)

Accessibility findings are automated checks against Web Content Accessibility Guidelines (WCAG) 2.1 and 2.2. They flag potential barriers and legal risk, not a certification or a determination of compliance with the ADA, Section 508, or EN 301 549. Automated testing catches only a subset of issues; a full conformance review needs manual and assistive-technology testing by a qualified reviewer.

FROM FINDING TO FIX

Want the fix, not just the finding?

Install Harvv and we turn each issue above into a ready-to-paste prompt for your AI coding assistant. Drop it into Cursor, Claude, or Copilot and the diagnosis becomes a concrete code change, written against this exact page.

Get the fix prompts

"How we know": **unlabeled** = a deterministic fact, identical on every load (e.g. element sizes). Most findings are this kind, so we only mark the exceptions: **median across loads** = a noisy lab metric, reported as a median. **real-user field data** = Google CrUX, actual Chrome visitors.

Structural and AI-search checks crawl up to 8 pages from your sitemap (a sample, not your full site). "Broken" means a link returned 404, 410, or 5xx, or did not respond; access-controlled pages (401, 403) are not counted.

02

Performance: phone, desktop, and real visitors

METRIC	MOBILE	DESKTOP	READ
TTFB (lab median)	10 ms	12 ms	LAB
FCP (lab median)	106 ms	152 ms	LAB
LCP (lab median)	232 ms	236 ms	GOOD
Page weight (median)	1.5 MB	1.5 MB	OK

Google Lighthouse (lab): Performance **74** mobile / **95** desktop, SEO **100**, Accessibility **80**, Best Practices **96**.

Lab numbers are from a headless mobile browser on an unthrottled connection: treat them as a floor, not a typical experience. Add a Google API key to light up real-user field data (CrUX) and Lighthouse scores.

03

Tiny buttons are hard to tap on mobile

14 of 40 tappable items on this page come in below the platform minimums for reliable tapping on a phone (Apple recommends 44pt, Android 48dp; WCAG 2.5.8 sets 24px as the hard accessibility floor). The same ones came up small on every one of the 4 test loads, so this is the page itself, not a fluke.

When customers can't tap what they expect to, they get frustrated and many of them leave. They don't file a bug. They don't try again. They just leave. A desktop dashboard can't see this because it's the difference between a thumb and a cursor.

The buttons measuring below the minimum on this scan:

- a 1x1 "Skip to content"
- button 11x11 "Slide 1"
- button 11x11 "Slide 2"
- button 11x11 "Slide 3"
- button 11x11 "Slide 4"
- button 11x11 (no visible label)
- button 11x11 (no visible label)
- button 11x11 (no visible label)

The fix is CSS-only on most sites: add padding around the icon (don't just change the icon size) so the actual tap area is at least 44×44 pixels. No redesign, no new assets.

04

Technical SEO & structured data

CHECK	RESULT
Title	Done-For-You SEO Blog Writing Service - Nina (44 chars)
Meta description	160 chars
H1	1 on page
Canonical	PRESENT
Structured data (JSON-LD)	1 block(s)
Open Graph	TITLE + IMAGE

05

The fix checklist

Everything to fix, priority first, each tagged with the screen it affects and a rough effort. Work top to bottom.

☐ Tiny buttons are hard to tap on mobile MOBILE CSS ONLY

☐ JavaScript crashed while the page was loading BOTH DEV AFTERNOON

☐ Images have no width or height set (layout is stable for now) BOTH CSS ONLY

☐ Unused JavaScript is being downloaded BOTH VARIES

☐ The page loads from a lot of third-party services BOTH DEV AFTERNOON

☐ 19 form fields have no label BOTH VARIES

☐ Meta Pixel (Facebook) is installed but misconfigured BOTH VARIES

☐ Add quotations so AI engines cite this page BOTH VARIES

☐ Broken internal links BOTH VARIES

☐ 6 potential dead-click targets BOTH CSS ONLY

☐ 2 inputs use the wrong type attribute BOTH VARIES

☐ Some text is low-contrast and hard to read BOTH VARIES

☐ Some buttons have no accessible name BOTH VARIES

☐ Security headers are missing or weak BOTH VARIES



14 interactive elements have no stable, accessible identity

BOTH

DEV AFTERNOON

Effort is a rough read from the outside: "CSS only" means no new assets or backend work, "1 line" means a single tag, "Dev afternoon" means a developer needs to touch tracking or scripts.

06

What this report cannot tell you

Everything above is from the outside, looking at the page on a simulated phone and desktop. The questions that actually decide revenue need real visitors. Install the Harvv pixel (one script tag, 16 KB, zero personal data, no engineering project) and within about 72 hours you'd know which buttons real customers tapped and missed, how often Google Analytics is missing visits, and exactly where mobile shoppers stalled and left. This report shows you where to look. The pixel shows you how often it happens, and to whom.

07

How we did this, and what it can't prove

- **4 mobile + 2 desktop loads** of one URL from headless Chrome (iPhone viewport at 390px, desktop at 1366px), August 30, 2026. Enough loads to separate real defects from random noise, not a full-site crawl.
- **Lab numbers, not real-user numbers** (no field data was available for this run). Real devices on real networks run slower.
- **Friction is inferred, not counted.** We can prove a button is small. We can't, from the outside, count how often it causes a missed tap. That requires the pixel on a live page.
- **This page rotates its content** load to load, which is on its own a reason a single-shot scan can't be the last word on it.

About Harvv, the source of this teardown

Harvv is a behavioral UX analytics platform (harvv.com). A lightweight JavaScript pixel captures how real visitors behave on a site (dead clicks, rage clicks, scroll depth, Core Web Vitals, JavaScript errors, and 50+ other signals) and the engine turns them into prioritized, plain-English findings. This teardown is the outside-in version of that: the same detectors run against a public page, with no pixel installed.

How to read it. Every finding here is a reproducible, automated measurement, not an opinion: element sizes, contrast ratios, load metrics, and structured-data checks that anyone can re-run against the same URL. The method is stated in full above. Automated testing catches a subset of issues, so this is a starting point, not a certification.

Full disclosure. Harvv makes the pixel that would measure the friction these findings imply, so we have a commercial interest. That is exactly why the findings are kept to things a reader can verify independently, and

why nothing here is inflated: an unreproducible claim would undermine the tool it is meant to demonstrate.

Prepared by Harvv (harvv.com), a behavioral UX analytics platform. Last updated August 30, 2026.