

Technical Writing Metrics That Actually Matter: An Overview



84% of developers learn primarily from docs; 90% rely on API/SDK docs daily



2026: 49% of docs teams track no internal metrics, down from 55% in 2025



Vanity metrics impress: page views, word counts, page counts—but show no answers found



Three tiers used here: content health, user behavior, business and developer outcomes



A metric matters when tied to a task, attributable to a decision and an owner



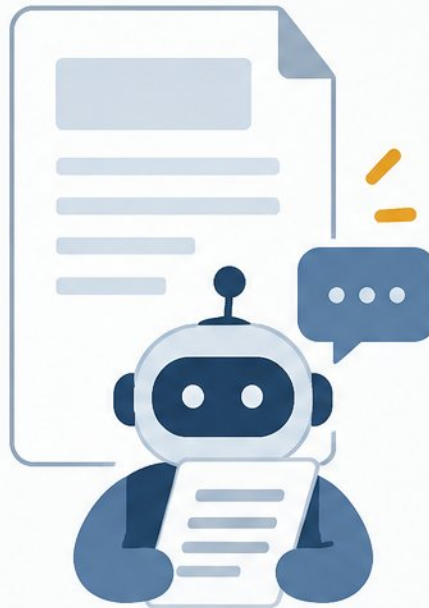
Why Most Documentation Metrics Fail

Popular Page, Little Understanding



High page views,
but no one gets answers.

Quieter Page, Real Impact



AI agents answer well,
even when page views
collapse.



Page views, time on page, and word count: most tracked, least explanatory



Tailwind CSS paradox: AI agents answered well, page views collapsed



Survivorship bias: only counts users who found the docs



High "Was this helpful?" scores can coexist with unresolved tasks



Context changes meaning: bounce on reference vs. getting-started guide



Attribution is hard: docs impact is tangled with UX and support



A good metric ties to a task, a decision, and a named owner

A Tiered Measurement Framework: Content Health, Behavior, and Outcomes



Tier 1, content health:

freshness, coverage, structure,
link integrity, terminology



Writers



Tier 2, user behavior:

search success, zero-result rate,
task completion, deflection



Leads



Tier 3, outcomes:

ticket reduction, time-to-first API call,
activation, retention



**Product,
Support**



Build a balanced scorecard; no single north-star docs metric exists



Keep tiers separate so missing content can't hide behind incorrect content

Content Health Metrics You Can Trust



Freshness checks: `last_reviewed` in frontmatter, max-age gates in CI



Coverage honesty: endpoint coverage alone inflates; add error code and example coverage



Structural checks: heading hierarchy, scannability, markdownlint alongside prose linting



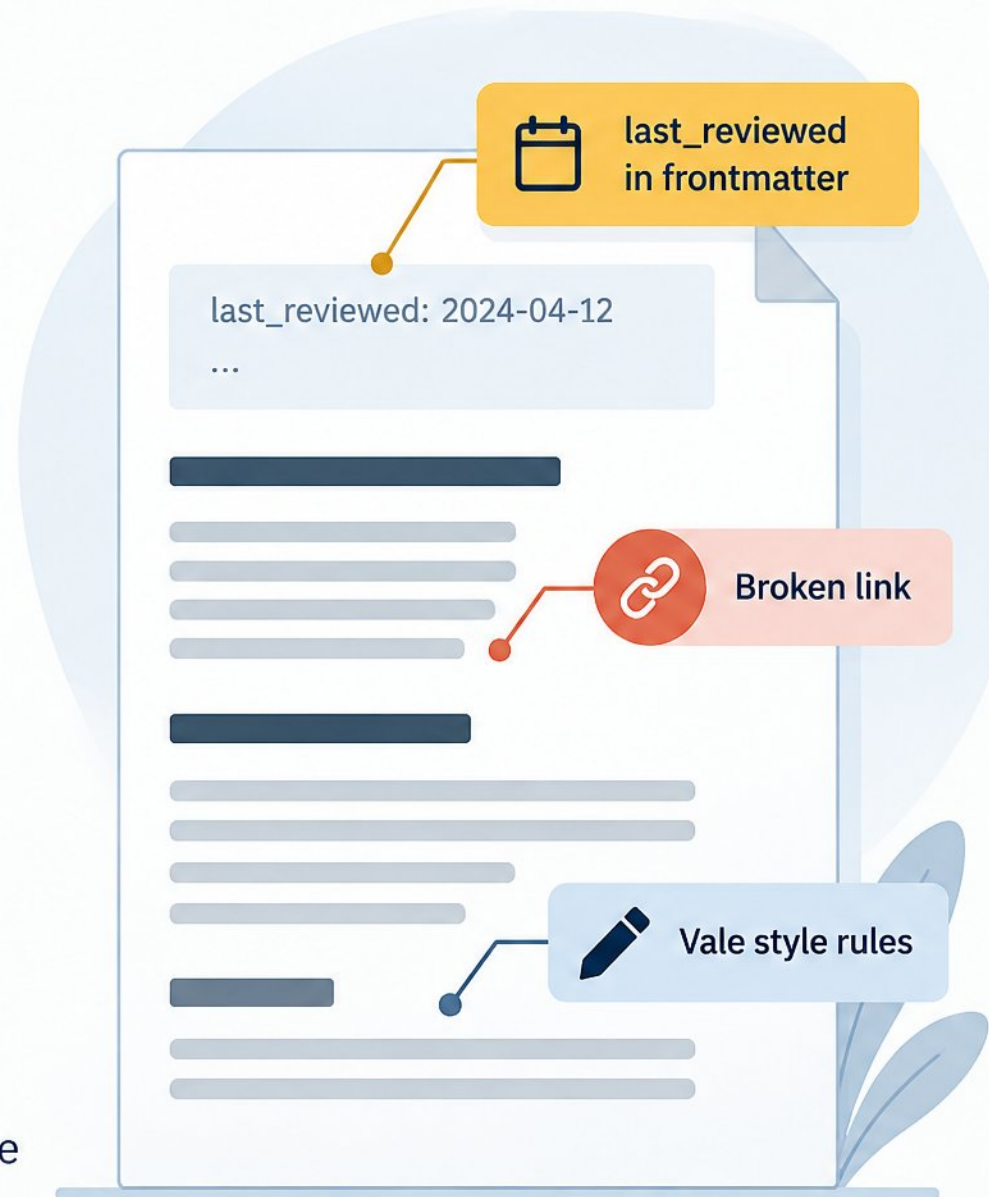
Link integrity: internal links on every PR; external URLs on a daily schedule



Vale style rules enforce terminology, tone, and front matter consistency



Set thresholds and review SLAs first, so stale pages get an owner and a due date



Behavioral Metrics: Search, Navigation, and Task Success



🔍 Zero-result search rate: under 8% mature, under 15% new; every miss is a content brief

🔑 Search followed by instant exit signals a title or index mismatch

🔗 Journey signals: copy-code events, step completions, back-buttons, dead ends

🕒 Time-to-answer: track the 15-second to 3-minute band, not raw average

📊 Separate true deflection from assisted containment; define the look-forward window

🛡️ Consent-gated analytics, no PII, server-side capture where feasible

💬 Pair numbers with one-question feedback and a few observed task sessions



**Turn behavior
into better content**

Outcome Metrics: Connecting Docs to Business and Developer Success



Time to First Successful API Call:
onboarding, reference, and sandbox
quality in one number



Ticket deflection delta per topic:
25-40% drops get docs funded



Integration success and SDK adoption
separate 'got it working' from 'kept using it'



Leading indicators: PR cycle time,
stale page count. Lagging: ticket volume,
activation



Use cohort or pre/post designs, not one
screenshot, to correlate docs with support load



State your measurement window, baseline,
confounds, and what would falsify the claim

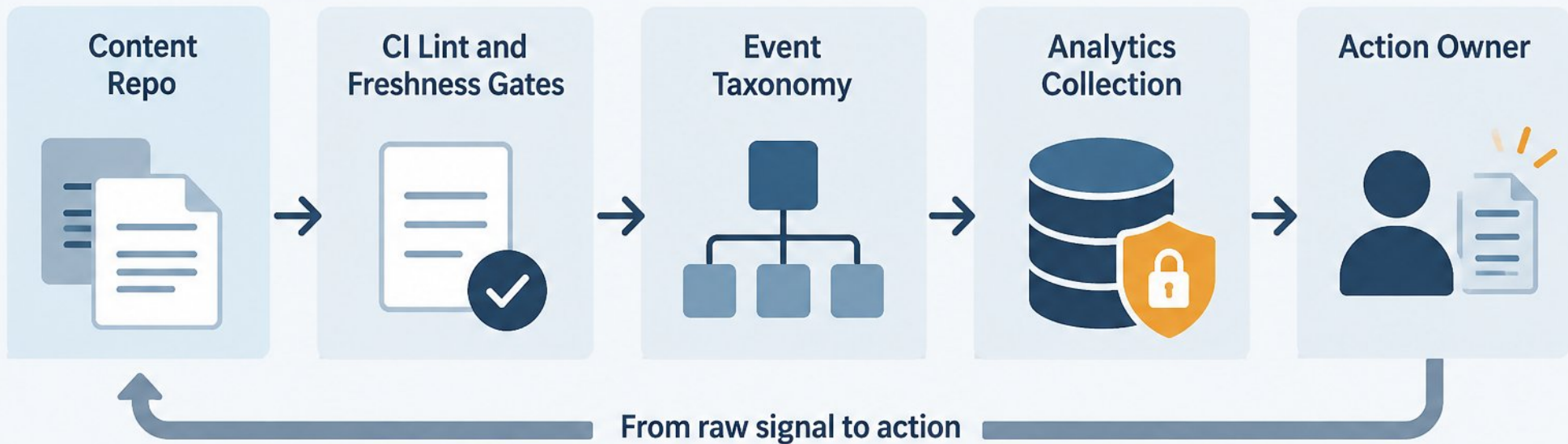
Developer Onboarding Journey



Support Tickets Trend

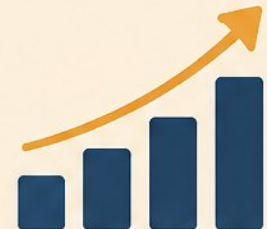


Instrumentation, Tooling, and Data Hygiene



- Analytics stack: page tags, search logging, feedback widgets, CI lint and freshness gates
- Define a small, stable event taxonomy before building any dashboard
- Lock event and property names early; renames break every funnel downstream
- One canonical content ID ties pages, snippets, retrieval chunks, and endpoints together
- Deduplicate by content ID and time window; document attribution rules
- Privacy: consent-gated tracking, allow and deny lists, aggregation thresholds
- Minimum viable setup: search logging, then a link crawl, then one feedback widget

Good instrumentation turns content signals into clear next steps.





Common Pitfalls and Ethical Considerations



Gaming metrics: page views and acceptance counts reward activity, not answers



Dashboards nobody reviews are decoration; every metric should point at a fix



Numbers show where friction is, rarely why it happens



No PII in event properties; aggregate small cohorts; state what is collected



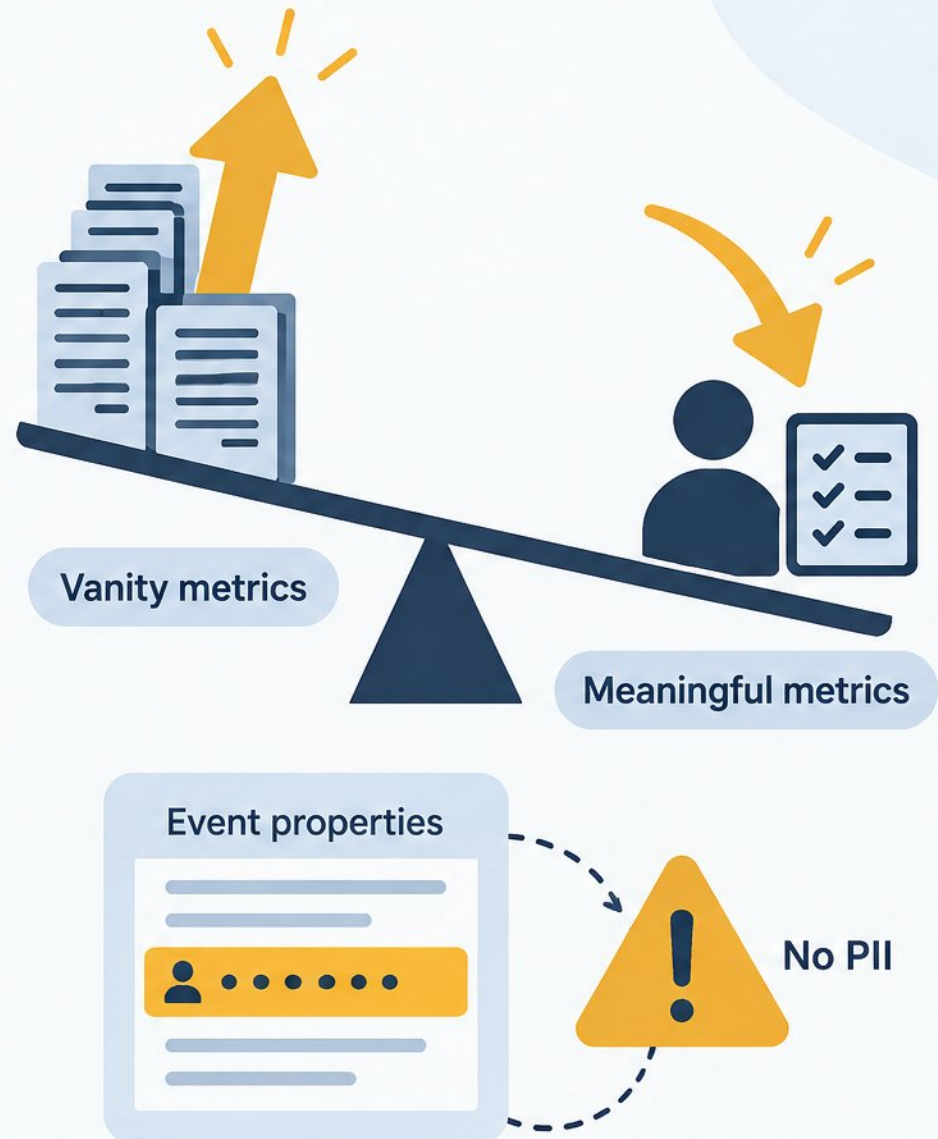
Never punish writers for product defects or page views on reference pages



Keep evaluation metrics contestable, with a correction path and access log



Read honestly: pair headline numbers with counterbalancing metrics



Putting It Into Practice: A 90-Day Measurement Plan, Weeks 1-4



Days 1-7

- Days 1-7: define business questions, name the executive sponsor, confirm your readers



Days 8-14

- Days 8-14: inventory content and map data sources across docs, analytics, support, CRM



Days 15-21

- Days 15-21: turn on search query logging and run a link crawl



Days 22-28

- Days 22-28: pick 10 high-risk pages or recurring questions as pilot scope



- Instrument only chosen paths: search events, one feedback widget, code-copy events



- State what you are deliberately not measuring yet, for a credible readout



Putting It Into Practice: A 90-Day Measurement Plan, Weeks 5-12



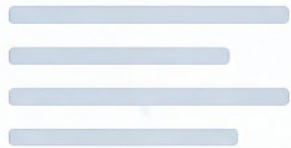
Iterate: retire metrics that never pointed at a fix; escalate those that repeatedly did



Embed cadence: book next review and baseline refresh before the pilot ends

Templates: Metric Definition Sheet, Dashboard Outline, and Stakeholder Update

Metric Definition Sheet



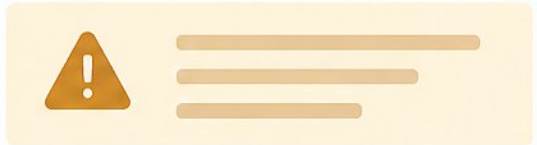
- Metric definition sheet: name, tier, definition, source, calculation, owner, baseline, target, cadence, limitations

Executive Dashboard



- Executive dashboard: one page on user success, friction points, monthly changes, next action
- Operational view: page entries, exits, task events, search terms, post-read actions, last-updated dates

Stakeholder Update



- Appendix: citation and deflection rules, attribution windows, deduplication, and what the numbers cannot prove
- Stakeholder update: risk or opportunity, evidence, decision needed, owner, verification signal



Keep it to one sheet, one dashboard, one recurring update.



Content health



User behavior



Outcomes

Adapting to AI-Mediated Documentation Use



Agents consume docs without page views, breaking traditional gap-detection loops



Track machine-readable access separately: lms.txt retrievals, raw Markdown, MCP availability



Layered retrieval metrics: retrieved, grounded, cited, user-visible



Run a coding assistant against your docs; verify the integration works



Validate AI summaries against source; hallucinated fields fail harder



Pair agent signals with task completion and ticket data



Key Takeaways and Your Next Three Actions

MEASURE → UNDERSTAND → IMPROVE

- A metric matters only if tied to a task, a decision, and an owner.
- Read tiers together: health shows what's broken, behavior where users struggle, outcomes why it matters.
- Start this week with search query logging and a link crawl.
- Set baselines before changing content; define thresholds of action upfront.
- Commit to one health, one behavior, and one outcome metric, each with an owner.

Your Three Commitments



Content Health

Owner:

Review Date:



Behavior

Owner:

Review Date:



Outcome

Owner:

Review Date:

PersonWise.

PersonWise • AI digital-human course platform



Scan the code or click anywhere to watch this course live, with voice Q&A

personwise.ai/t/e901737c/public