Changelog: Kaloko

0.22.0

Released 2026-10-20

Added

• A Health checks page in the help lists every pack and each of its checks: what it looks at, why it matters, its severity and where it applies, in English and Czech. It is written from the same code the CLI runs, and the Health tab on the canvas links to it.

• The scenario page in the app shows which health packs the latest run had on and off, and how to switch one with packs: in the scenario file.

• Light or dark, your pick: the account menu in the app and on the canvas has a theme switch (System, Light, Dark). The choice stays on this device and applies before the page paints, so nothing flashes white at night.

• The account menu holds who you are, the language of Kaloko, the theme, a link to the Kaloko website and Sign out. On a phone the same items sit at the bottom of the menu.

• Signed-in people reach the website again: "Kaloko website" in the account menu and the footer leads to the home page instead of back to your organization.

• A product you no longer ship can be archived: kaloko products remove <product>, or Archive product at the end of its changelog settings. It disappears from Products, its changelog and docs answer "not found", and its releases and docs stay stored. kaloko products sync reports it as archived instead of creating it again, and kaloko products restore <product> brings it back.

• The help page on browsers and devices describes the default capture setup and how to change it.

• kaloko scenario move <scenario-id> --project <slug> moves a scenario to another project with all its runs, design drafts and screens. Admins do the same in the app, in Scenario settings on the scenario's page, after a confirmation. The move wins over project: in the config, so later shares land in the new project too.

Changed

• On a design environment the code-only packs (seo, aeo, perf, console, errors) are off, because a designer's HTML has no real server, meta tags or error statuses. health: { design_packs: [console] } in the scenario keeps one on. The run says which packs it turned off (packs_off in run.json).

• Design versions are listed under Design drafts on the home page and no longer among the runs.

• The canvas explains Live, Inspect and Dev mode in a line each: Live runs your own uploaded HTML, Inspect styles shows an element's styles and tokens, and Dev mode is the handoff to development. Kaloko does not write code from a design.

• The canvas shows the same main links as the app, in the same order: Scenarios, Runs, Screens, Projects, Products.

• The language of Kaloko moved from the canvas bar and the app header into the account menu, so it no longer sits next to the run's own locale switch.

• The menu button on a phone is round with three lines in the app too, like on the canvas.

• Someone who is a guest or a viewer in another organization can now create their own. Only a full member elsewhere (Designer, Developer / Tester or Admin) is asked to sign in instead. Sign-in goes to the organization where you are a full member first; the others are in the organization switcher.

• New scenarios from kaloko init, kaloko draft and the template capture mobile at 360 × 800 instead of 390 × 844, next to desktop at 1280 × 800. Existing scenarios keep their sizes and baselines. A tablet is added only when you ask for one (tablet: [744, 1133]).

• The scenario and runs lists no longer look the same. A scenario row starts with its trend, the number of runs and the last result; the runs list is split by day, and each run starts with its time and who shared it.

• Branch and commit moved from their own column to a grey line under the scenario name, with a long branch cut short (hover shows it whole). Both lists now fit a 1280 px screen without a sideways scrollbar.

• The scenario list shows the project as a plain label. The project picker in every row is gone, so a stray click no longer moves a scenario with all its runs.

• Return and Approve on a step have their own bar under the comment box, so a comment no longer sits one click away from a decision. In review mode you decide in the bottom bar only; the side panel is for comments.

• In Live mode the comment box says that comments go on an element of the page and starts picking one; pinning and drawing stay on the screenshot.

• "Mark a changing region" no longer disappears on drafts, for viewers and reviewers, or on a step without a screenshot. The button stays, disabled, and says why.

Fixed

• The Screens page follows dark mode: tiles, picture frames and the status labels used light colours on a dark page.

• On the Screens page, the "not checked" label no longer hides off screen.

• The canvas viewport switch offers only the viewports the run has: no "Both" for a run captured on one, and Tablet when the run has a tablet (with three or more viewports the first choice is "All").

• The gate of an environment (HTTP Basic, service-token headers) no longer follows a redirect to another host. Podcast audio that a gated page sends on to S3 now plays instead of failing with HTTP 400, and an identity provider or a link checked on another host never sees the gate. Redirects within the gate keep it, as before.

0.21.3

Released 2026-10-15

Fixed

• In a step's detail, opening an AI-evaluated or failed criterion no longer squeezes its explanation into a narrow column with the technical detail spilling over it; everything sits under the criterion's title again.

0.21.2

Released 2026-10-13

Changed

• Stack pages now report console errors. Before, the check passed on every page, so stack steps with no_console_errors or the console pack may fail after this update where they passed.

• kaloko doctor --json now lists the token's scopes in the token check, so a CI script can check that the token may make a run public (full), not only share it (upload), before it spends twenty minutes walking.

Fixed

• Console errors on the shared browser stay with the capture they belong to. A capture used to list every error the browser had seen on the same page, so one failed request from yesterday's test showed up in every run after it. Now kaloko capture counts only errors from the tab it captures, logged after the run started and after the last capture on that tab, and a walk counts what a page logged since the previous step on it was captured.

• The shared browser's console log starts empty with every kaloko browser start and keeps only its last few megabytes, so a browser left running for a week no longer piles up a log nobody reads.

0.21.1

Released 2026-10-11

Added

• Findings can open issues in GitHub Issues and Plane, not only in Jira and Linear. Connect them in Settings -> Integrations (the Kaloko GitHub app with the Issues permission, or a fine-grained token; a Plane API key), then

name the repository or the Plane project in PUT …/scenarios/<id>/findings/sync. One issue per finding, a comment when the number of pages changes a lot, closed when it is fixed and reopened when it comes back. Admins list which repositories and Plane projects each Kaloko project may send findings to.

• Routes per check or pack in the findings sync: send canonical, hreflang and structured data to the developers' repository and thin content or low click-through to the marketers' Plane project, from one scenario.

• With traffic: on the stack, findings carry the clicks, impressions and impact of their pages, and so do the issues the sync opens.

Fixed

• A tracker that timed out while creating an issue for a finding could get a second one on the next run. The finding now waits and says the issue may exist; after a look at the tracker, Try again on the findings page lets Kaloko create it with the next run.

• Evaluating the same run again no longer sends a step's recording to triage a second time. On a busy machine the first sampled frame sometimes came out empty, so the second evaluation picked other frames and missed the cache. Kaloko now waits until a frame is really drawn and remembers the frames it chose for each recording, so they are the same every time.

0.21.0

Released 2026-10-11

Changed

• Stack pages are now captured whole by default, like every other step; until now a stack kept the first screen only. This is a changed default, and stack runs grow several times: a typical long page stores a screenshot of about 600 KB instead of about 70 KB, and a thumbnail of about 180 KB instead of about 25 KB, because the thumbnail is as tall as the page. Vision checks now see the whole page too, not just what fits on the first screen. To keep the old size, set capture: { full_page: false } on the stack; capture: { screenshot: thumb, full_page: false } is the smallest and suits big monitors that need the page facts rather than the picture.

• When a stack captured whole takes most of a run close to the per-run limit, kaloko share names it and suggests full_page: false; a run over the limit lists full_page: false among the ways to trim it.

• Captures finish finite animations, and most captures show endless ones (a spinner, a pulsing badge) at their first frame. On pages with animations the first run after the update can differ from the runs before it, even though the page did not change.

Fixed

• Before a capture Kaloko waits until what scrolling changed on fixed and sticky parts is back as it was at the top (a header that floats while scrolling, a search panel fading in), or has stopped changing for a second, at most 3 s. Full-page captures no longer show a sticky header caught in its floating state with a half-visible panel.

• A full-page capture no longer catches CSS transitions halfway: resizing the page for the capture restarted them on some sites (a search panel fading out again over the breadcrumbs). Transitions end at once and finite animations finish for the capture.

• A page of a stack has its own Health tab, as a step has: its health checks with the evidence for that page, findings first, and Adopt for the stack step. Before, health checks of stack pages were only in the run's Health panel and the page listed them among its acceptance criteria.

0.20.1

Released 2026-10-11

Added

• Hosted runs group the language versions of a stack page by hreflang (items.variants: hreflang), as kaloko walk does: the same pages in the same week in every language, one row per page in the stack index, and a missing translation marked on the page and counted in HC1005. Reading a source page for its hreflang counts as one

hosted page. kaloko validate no longer warns about such stacks in hosted runs.

Fixed

• kaloko walk reads the sitemaps and robots.txt of a stack with the environment's access (its headers, HTTP Basic, client certificates and the walk's cookies), so a staging behind a password no longer ends with an empty stack and HTTP 401. A sitemap on another host, or one that redirects there, is still read without them. Each file has 60 seconds and 60 MB.

0.20.0

Released 2026-10-10

Added

• Findings: what a scenario has open in health is grouped by check, site, language and step, so two hundred product pages without structured data in Polish are one row, not two hundred. Each finding has its severity, how many pages it hits, up to ten of them with the CMS edit link, and when it first and last appeared. Health -> Findings shows the table per scenario; GET …/scenarios/<id>/findings and MCP list_findings give the same to dashboards and agents (a read:health token reads them too).

• Acknowledge a finding when someone is on it, or ignore it with a reason. The decision stays with the finding from week to week; a finding is fixed once none of its pages fails. Anyone who may comment on the scenario may decide, in the app or with PATCH …/scenarios/<id>/findings/<id>.

• A scenario can open a Jira or Linear issue for each new finding at or above a severity, comment when the number of pages changes a lot, close it when the finding is fixed and open it again when it comes back (PUT …/scenarios/<id>/findings/sync, Developer / Tester). A run opens at most 20 issues; the rest follow with the next run.

• Hosted runs read stack steps: pages listed by items.urls, a sitemap (gzipped and sitemap indexes too) or the sitemaps in robots.txt, with match, exclude, limit and rotate: week picking the same pages as kaloko walk. Each page is read without a screenshot: checks of selectors, texts, titles and meta tags are decided, index fields such as title, H1, description and JSON-LD types are filled in, and the canvas shows the stack with its index. Big stacks are read over several quarter-hour rounds, at most 1,000 pages per run; Run now, deploy triggers and edits wait for a run under way.

• kaloko schedule add takes an environment with a base_url per language; the hosted run opens each language at its own address, otherwise at default, as the walk does.

• Settings -> Integrations and kaloko schedule list show how far a stack run got and how many health results the last run found new and fixed.

• items: { variants: hreflang, source_locale: cs } on a stack step shows the translations of a page as its language versions instead of separate pages. The list, limit and rotate come from the source language, and every other language walks each page's own hreflang alternate, so the stack index has one row per page with a screenshot per language and all languages check the same pages in a given week.

• A page without a translation is reported instead of left out: the stack index marks the missing language, HC1005 counts the pages (no hreflang for the language, only x-default, a hreflang back to the page itself, a translation that answers an error, or several pages sharing one translation), and kaloko export findings lists one row per page and language.

• Traffic data for the fix list: traffic: in kaloko.config.yml or the scenario joins Search Console and GA4 numbers onto the walked pages, from a CSV export (English or Czech headers) or, with a service account key, straight from the Search Console and GA4 APIs, once per run. kaloko export findings then adds clicks, impressions, CTR, position, sessions and an impact score and puts the most seen pages first, the page index on the canvas gets Impressions and Clicks columns, and the seo pack adds HC127 for pages that are seen often in search but clicked less than half as often as the site's pages at similar positions. Nothing is requested unless you configure it.

Changed

• Every page of a stack in a hosted run counts against the monthly budget of hosted pages. When the budget runs out halfway, the run waits up to a day for more, then shares the pages read so far and names the rest as not walked.

• A hosted run with stack steps takes up to 30 languages (10 without).

Security

• Sitemaps and robots.txt read by hosted runs go through the same public-address check as pages, on every redirect and for sitemaps on other hosts too, without the environment's credentials; pages are opened only on the environment's own addresses. match and exclude are checked when a schedule is saved and run in a linear-time matcher.

0.19.0

Released 2026-10-10

Added

• Check nojs_parity: one more request per page fetches the raw HTML without JavaScript and compares it with the rendered page. It fails when the title or canonical differ, when the meta description, H1 or a JSON-LD type appear only after scripts run, or when the raw body text is below 30 % of the rendered one (ratio:). With user_agent: gptbot (or oai-searchbot, claudebot, perplexitybot, googlebot, bingbot, a whole user agent string) it comes like that crawler, without cookies, and a page robots.txt closes to it is reported as blocked, not fetched. A bot the site, a WAF or a gate turns away (403, 429, a challenge page, a redirect to a sign-in) is not evaluated, with a note saying who was turned away. The aeo pack adds it as HC905 for GPTBot. Hosted runs show it as not evaluated; kaloko recheck keeps the walk's result.

• An evaluator budget for large sites: evaluate.budget.calls in kaloko.config.yml, or kaloko evaluate --budget N and kaloko walk --budget N, caps the paid JEV calls of one evaluation. Ordinary steps are asked first, then the pages of each stack, starting with pages that fail a deterministic check and pages checked longest ago. What does not fit stays "not evaluated" with a note, and kaloko evaluate says how many captures it left out. Cached answers cost nothing, so the next evaluation picks up the rest. The same captures always spend the budget on the same pages.

• items.semantic_order: priority on a stack step picks its semantic sample the same way without a budget; by default the sample stays the first pages in walk order.

• Your own text evaluator behind an OpenAI-compatible API, such as Ollama (http://localhost:11434/v1) or LM Studio: an evaluator of kind: openai_compatible with base_url and model, chosen with evaluators.text. It answers the same questions from the same page text. With evaluators.escalate_to: jev, answers it is not sure about go on to JEV, counted in the budget. Its verdicts are marked as your own model's on the canvas, in acceptance records and in the pull request comment. Nothing changes until you configure it, and the page text goes only to your base_url.

• A dashboard can read the health of a run already counted: GET …/runs/<id>/health/aggregate?by=locale,step,check gives pass, fail, unsure and not evaluated per language, step, check or pack, with the pages hit most. Group ids stay the same from week to week. GET …/scenarios/<id>/latest finds the newest shared run of a scenario (?environment=), and …/latest/health/aggregate answers for it in one call. MCP has get_health_aggregate.

• Health over time: GET …/runs/<id>/health/diff (MCP get_health_diff) lists, page by page, what fails or is unsure now and did not in the previous run of the scenario on the same environment, what was fixed and what is still open, each with the run where it first appeared. A page the run did not check (a rotating sample, a language left out) stays open as unchecked instead of counting as fixed. The health page shows new and fixed per scenario, and the scenario's health history has them per run.

• The health.changed webhook tells you after a shared run when something in its health is new or fixed, with the counts and the first results.

• Tokens with the read:health scope read health and the newest run of each scenario and nothing else, for a dashboard that pulls once a week. Create one in Settings -> API tokens (on Enterprise also as a service

account). The fields are described in the new help page "Health API for dashboards".

• kaloko init --seo <site> drafts a weekly SEO and answer-engine monitor from the live site: robots.txt and the sitemaps give the languages, a sample of each language's pages the biggest content families (first path segment), paired across languages by hreflang, the same segment or size. It writes qa/flows/seo-<site>.yml with one rotating stack step per family (match per language, rotate: week, packs seo and aeo; a11y and perf with --packs), the production environment in a new config, and prints the next steps. A site that cannot be reached writes nothing; a site whose pages live on its www (or https) name is followed there.

• Stack pages can link back to their source: an item script or a JSON list gives edit_url and content_ref, or items.resolve: steps/resolve.mjs looks them up for the walked pages (15 s per page). The page detail shows "Edit in CMS", and the stack CSV and MCP get_stack_items carry both.

• kaloko export findings [--format csv|json] [--locale l] [--step s] [--out file]: the fix list of a local run, one row per failing criterion or health check, page and language, with the evidence, edit_url, content_ref and the command that walks that page again. It reads the run and writes nothing into it.

• Pack aeo (answer engines: AI search and assistants), advisory like the other packs: HC901 questions and answers on the page (details/summary, dl/dt, two or more headings ending with "?") come with FAQPage or QAPage structured data, HC902 a short answer-first paragraph follows the H1, HC903 /llms.txt answers at the site root (one request per site and walk), HC904 a meta description is present. Pages without questions show HC901 as not evaluated. Hosted runs check HC904 and mark the others "not evaluated" with one note.

• Stack steps read gzipped sitemaps (sitemap: /sitemaps/{locale}/sitemap.xml.gz, or gzip bytes behind any URL) and follow sitemap indexes two levels deep.

• items: { sitemap: robots } (or robots: true) takes the pages from the sitemaps listed in /robots.txt. When their URLs name the walked locale (/sitemaps/de/…), only that locale's sitemaps are read. match, exclude and limit work as before.

• Stack index fields jsonld_types (the structured data types of each page, for example Product, Offer, BreadcrumbList) and meta tags by name, such as from: og:title, from: twitter:card or from: meta:robots, so items.json carries them for every URL.

• items: { limit: 100, rotate: week } walks a big catalogue in turns: each ISO week (or each run with rotate: run) takes the next limit pages of the list, so every page is checked within a few weeks. The stack index shows per language which part was walked and how many pages were never checked, and each page its last_checked.

• match and exclude of a stack step take a map by language ({ default: '^/eshop/', fi: '^/fi/tuotteet/' }) for sites that translate their URL segments.

• An item script gets the step (node: id, title, lane, session) and items.params, so one script serves several stacks.

• expect_status: 404 on a step says it is an error page. kaloko draft sets it on the drafted 404 step.

• The seo pack checks structured data: HC112 JSON-LD is on the page, HC113 every JSON-LD block is valid JSON (a warning names the block and the parse error). New assert jsonld_present takes types: [Product, BreadcrumbList] for pages that must carry certain types; jsonld_valid is the parse check.

• Every capture records structured_data (blocks, invalid blocks, the schema.org types and the recognized ones: Product, Offer, BreadcrumbList, FAQPage, Article, BlogPosting, Organization, WebSite and a few more). A stack step's items.json and its manifest count pages per type, pages without any structured data and pages with invalid JSON-LD, so a dashboard can chart them week by week.

• A stack step with the seo pack now compares its pages with each other after the walk: shared titles, meta descriptions and H1s within one language (HC121–HC123), canonicals that point elsewhere or to a page that answers an error or says noindex (HC124), hreflang the other walked language version does not answer (HC125), and sitemap pages that answer other than 200, redirect or say noindex (HC126). Advisory like the pack; the findings count pages, and items.json lists the affected addresses in groups.

• Pack locale for multilingual sites: html lang, the page's own hreflang, the language of the text and the decimal separator of prices are checked against the walked locale (HC1001–HC1004). An untranslated page in a language version shows up as HC1003.

• Site checks with health: { site: true } in the scenario, once per site and run: robots.txt parses and does not block the walked start pages (an error on production, a warning elsewhere) or AI crawlers, every sitemap it lists answers and is valid with absolute same-host addresses and sane lastmod dates, /llms.txt is there, and http:// and the other host name (with or without www) redirect in one hop (HC1101–HC1107). kaloko health and the canvas show a row per site; health.site.environments: [production] keeps staging walks from checking.

• capture: { screenshot: thumb } keeps only the 640 px thumbnail of each page as the step's picture: for stacks of hundreds of pages whose checks read the page, not the picture (SEO crawls).

Changed

• kaloko walk --items picks stack pages from every matching page, not only from this week's rotation window, so a fixed page can be walked again in any week; such a walk leaves the rotation state as it was.

• match and exclude apply before limit for every source of a stack step, scripts included.

• The errors pack checks HC601 (soft 404) and HC605 (the explanation) only on error pages: a step with expect_status of 400 or more, or a page that answered 4xx/5xx. On normal pages they are not applicable instead of failing. kaloko validate warns when the pack is on a stack step.

• kaloko validate and kaloko start warn when the config asks every scenario for its own project (a comment, or a product with projects:) and the scenario has no project:.

Fixed

• When kaloko share gets HTTP 404, it says why: which project the run names and which projects the organization has, or that the token belongs to another organization or expired (kaloko login --org <org>). Other commands say the same for a bare "Not found" on an organization or project.

• Changing the Kaloko app's repository access or permissions on GitHub now brings you back to Settings with the repositories read again, instead of an error saying the installation link expired.

• kaloko share checks the run against its limit by the size the service stores (HTML and JSON gzip-compressed), not the size on disk, and when a run is too big it names the largest steps. A 20-language SEO run of 1 400 pages that was refused at 529 MB is about a third of that as stored.

• The personal data check before sharing no longer reads the user part of a URL as an e-mail address (a public Sentry DSN such as https://key@o1.ingest.sentry.io/23 in a meta tag).

0.18.0

Released 2026-10-10

Added

• Every pull request in a repository connected to the Kaloko GitHub App gets one Kaloko check on its newest commit. It waits for the runs shared for that commit, from a coding agent in the cloud, your machine or CI, and turns green when they pass or when the change reaches no scenario. It keeps a stable name, so branch protection can require it before the merge. An optional Kaloko review check follows the decision on those runs.

• Modes per organization and repository in Settings -> Integrations -> GitHub: off, manual, automatic (the default) and strict. The github: section of kaloko.config.yml on the base branch can make the check stricter, never looser.

• One pull request can be controlled with the labels kaloko:skip, kaloko:all, kaloko:review and kaloko:run, with the comments /kaloko skip <reason>, /kaloko walk and /kaloko all, or with the buttons on the check. Only people with the access you choose may skip, never apps or bots, and the check names who skipped it and why.

• kaloko ci plan tells the check which scenarios a commit reaches, following imports like kaloko affected. The workflow from kaloko ci github sends it before walking.

• Settings list the installation's repositories with their mode, last check and what went wrong, in words you can act on, such as a repository that is not in the installation yet.

Changed

• A shared run carries its repository, the project's folder in a monorepo, and whether tracked files differed from

the commit. The pull request is found by commit, so a sandbox without gh still gets its check. A run of uncommitted changes does not count for the commit.

Fixed

• A link to Kaloko from another site (a portal, GitLab, Teams, an e-mail) opens the page for someone who is signed in, instead of asking them to sign in again.

• A link to a step (#step=…, #lane=…, #session=…) still opens that step after signing in.

• Links to Kaloko's public changelog and help now show a preview picture in chat apps and social posts.

• In replay, -> and the Next button follow the diagram: the next screen on the same row, forward before back, instead of whichever connection the run happened to list first.

• Open index on a stack in replay shows the stack's pages; before, they opened hidden behind the replay.

• A step reference shown as an example inside a code block or inline code is no longer resolved as a picture, so kaloko docs check and kaloko docs publish stop warning about examples.

0.17.0

Released 2026-10-09

Added

• Webhooks now tell you about comments and review changes: comment.added, comment.resolved, comment.changed, review.step_changed and run.returned. A QMS portal can show the open comments of a run next to its test cases and link straight to the thread, without polling.

• Each webhook takes the projects you tick, like a Slack or Teams channel. A restricted project's events reach only the webhooks that name it.

• Settings -> Integrations -> Webhooks lists each webhook's recent deliveries. A delivery that fails is tried again for about a day with the same event id. Events, payloads and signature checks are in the help under Webhooks.

• A documentation portal can show a step's screenshot next to its test step: GET /api/v1/orgs/<org>/runs/<run>/steps/<step>/screen with a read token answers the full screenshot or its thumbnail with its SHA-256. Pass the hash you stored as sha256 and a different file answers 409 instead of a newer picture. MCP agents get the same through get_step_screen.

• A run can name the records it serves outside Kaloko, such as a change request in your QMS: kaloko start --ref CC-2026-001 (repeat it for more, or set KALOKO_REFS in CI), and why it was walked with --purpose dry_run, formal or smoke (KALOKO_PURPOSE). The canvas shows both in the run's header.

• GET /api/v1/orgs/<org>/runs?ref=CC-2026-001 lists every run of a record with its verdict, environment and purpose; ?purpose=formal keeps the formal ones. kaloko runs --ref, the MCP tool list_runs and the evidence of a release take the same filters, and the evidence of a run carries run.refs and run.purpose.

• The trace matrix says who performs each step and in which part of the process: every row of trace.csv, trace.json and the evidence names the step's session (the role) and its lane (the sub-process). The evidence and MCP get_run_summary also list the flow's lanes and sessions, so a portal can show them without reading the YAML.

• A canvas link ending in #lane=1 opens the map on that lane with its band highlighted, and #session=qc brings one person's steps forward across all lanes.

Changed

• An invitation code now starts its trial the moment you sign up with it or enter it under Plan & billing. There is no checkout and no card. Plan & billing shows when the trial ends and has an Add a card button. Billing starts only when the trial ends; without a card the organization goes back to Free that day and keeps its runs.

• Organizations that already had the sample run get the current one (with health, requirement references and detail crops) in the same language, once; an organization that deleted its sample does not get it back.

• The sample run in a new organization shows what Kaloko does today: Health finds three small things in the tea shop (a picture without alt text, a missing meta description, a pale note under the delivery options), steps and

criteria carry requirement references, the cart and the payment have crops for user guides, and the keyboard walk leaves focus shots. The guided first review has one more hint, about Health, and points at the requirement and the crop on the way.

Fixed

• Organizations that entered an invitation code but never finished the checkout that followed were left on Free. Their trial now starts for the full length of the code.

• Dragging the map to pan no longer leaves lane headings, step titles and thumbnails highlighted as selected text. Text in the side panels, comments and fields still selects as before.

• On the page that approves kaloko login, picking another organization now reloads its projects, scopes and token lengths, so a project-limited token only ever lists projects of the organization it is for.

• Screenshots of runs shared with the deduplicated upload no longer go missing after an internal storage recount. The recount now leaves every file that points at stored content alone, and unreferenced content is kept for a week before it is cleaned up, so a lost reference can be restored.

0.16.1

Released 2026-10-07

Added

• When you type a command, sub-command or option your kaloko does not know but a newer release has, the CLI says which version has it and how to update (npm i -D kaloko@latest, or npx kaloko@latest …) instead of guessing a typo. It asks kaloko.app/cli/latest.json at most once a day, waits at most 1.5 s and stays quiet offline; KALOKO_NO_UPDATE_CHECK=1 turns it off. kaloko doctor shows when a newer kaloko is available.

• kaloko help --json lists every command with its sub-commands and options, for agents and scripts.

0.16.0

Released 2026-10-07

Added

• Signing in with Google, Microsoft or GitHub when your e-mail has no organization yet now leads to creating one: name it and you are in as its admin, without waiting for an e-mail link. The sign-up page offers the same buttons, and a trial code in its link carries over.

• Trial code links can point to the homepage: https://kaloko.app/?kod=CODE. The visitor first sees what Kaloko is, a slim banner on every page says what the code gives ("You have an invitation: 60 days of Business free, no card."), and the code stays with them until they create an organization, by e-mail or with Google, Microsoft or GitHub. An organization without a subscription starts the offer from Plan & billing with one click.

• Projects from the CLI: kaloko projects create <slug> --name "ECO Podcasty", kaloko projects rename <slug> --name … and kaloko projects sync, all with --json. The config takes project_name: (the name people see) and product: (a product the project is part of); kaloko init --project … --project-name … --product … writes them and, when you are signed in, creates the project on Kaloko. The MCP tool upsert_project does the same for agents.

• kaloko init in a repository with several sites or apps (a monorepo, Rails engines, several Folio sites) lists them instead of guessing routes. --site picks one, --base-url reads its pages from the running site (sitemap, then the links of its home page) and --skeleton writes a scenario with a TODO to fill in.

• Details for guides: name the parts of a screen a guide talks about under details: on a step (an id, a name and a selector). Every walk takes each one in every language and screen size with its section or card around it and a ring around the element, with personal data hidden as on the screenshot. In docs and the changelog, shows it; kaloko docs check warns about a detail the scenario does not declare or no accepted run has yet, exports carry the picture and the MCP tool resolve_step_image takes the detail id.

Changed

• New organizations cannot take an address Kaloko uses or keeps for itself: its routes (hooks, scim, …), words

for likely future pages in English and Czech, two-letter language codes and look-alikes of Kaloko, QAwalk and Sinfin. Existing organizations keep their addresses.

• The first kaloko share of a new project creates it with project_name: from the config, not with its slug. An existing project is never renamed by a share: when the names differ, the share says so and points to kaloko projects sync.

• kaloko init leaves out what is not a page: favicons and other static files, ad and tracking endpoints, API, admin and sign-in routes, routes that need a parameter and, with a running site, anything that does not answer HTML. The drafted scenario says at the top where its pages came from and what was skipped.

• The admin of a project can rename it with their API token (kaloko projects rename); everything else about a project (visibility, people, removal) stays in the app.

• Teammates need this CLI version or newer once kaloko.config.yml has project_name: or product:: older ones reject keys they do not know.

• The focus shots of the keyboard walk show on the Health tab under the focus check and beside the tab order of the shown language and screen. In Step attachments, which are now grouped into details, keyboard focus and crops, each focus shot says "Keyboard focus · HC309" and opens that check.

• The scenario browser leads with the scenario's name (up to two lines) and shows its id as a detail; the runs header names the scenario; the kind reads Task/Flow (Úkol/Průchod).

• A run opens in the reader's language when it has screens in it (an English reader sees the English screens first), and that language leads the locale list; an embedded canvas always follows its page.

Fixed

• Design drafts on the scenarios page follow its filters: a project shows only its own drafts, run filters leave them out, and each draft shows its name and project.

0.15.0

Released 2026-10-07

Added

• Requirement references: refs: on a scenario, a step or a criterion (refs: [URS-12, FS-3.4], free text). They reach the run, the acceptance record and the trace matrix. Runs also record the SHA-256 of the scenario file as walked and the CLI, Node.js, operating system and browser versions.

• Evidence for your own documentation (Business): for any shared run, accepted or not, and for a release. It says whether the run was accepted and whether it was signed in Kaloko, and lists the criteria as walked with who assessed each result (a person, a deterministic check or a named AI model, AI always advisory), screenshot hashes, decisions, signatures, the audit trail and the trace matrix. kaloko evidence <run> | --release <version> (--json, --csv), the MCP tool get_evidence, the API (…/runs/<id>/evidence, …/trace.csv, …/audit.csv, …/evidence?product=&version=) and the format qawalk.evidence.v1 with a JSON schema.

• The evidence package: kaloko signoff <run> --package or Evidence package (ZIP) on the canvas downloads the signed record (JSON and PDF), the evidence, the trace matrix, the audit trail and the screenshots with a signed list of every file. kaloko signoff --verify checks a package as well as records.

• Electronic signatures, optional and off by default (Business): an admin turns them on in Settings -> Security for every project or some. Accepting a run (and, if chosen, approving a step) then asks for the meaning of the signature from the organization's own list and a fresh second step; the record shows the printed name, e-mail, time with time zone and meaning. Changing a signed decision asks for a reason and the signature stays in the record as withdrawn. Tokens and agents never sign.

• The webhook event evidence.ready: a run's evidence is complete (shared, accepted or returned) or a release was recorded, with the state and where to fetch the evidence and the package.

• The acceptance record and its PDF list signatures, traceability, tools and environment, the AI models behind results, and advisory results such as health checks.

• Help: Kaloko in regulated work (GxP) and a supplier summary for quality assurance.

• kaloko health lists the health findings of a run by pack and step, with what is new or fixed since the previous run and how to fix each check; --json for agents, --remote for the shared run, --fail-on error for a team that wants health to stop a pipeline.

• Health checks have their own ids: HC101…HC705 (HC1xx seo, HC2xx usability, HC3xx a11y, HC4xx perf, HC5xx console, HC6xx errors, HC7xx app). Acceptance criteria stay AC. The number is the check's, so a pack kept in acceptance adds AC302 where the health check is HC302. Runs and scenario files written before keep working: health checks stored as AC302 show as HC302, the old ids work in kaloko health adopt, MCP and links, nothing stored is rewritten, and captures stay current.

• kaloko health adopt <step>:<check> [--text "…"] writes a health check into the scenario as an acceptance criterion of the step, as it is or in your words, so it counts from the next walk: home:HC103 becomes AC103 with from: HC103, and the step's health stops listing HC103. On the canvas, Adopt as criterion does the same, and kaloko health pull brings those adoptions into the scenario file.

• Health on the canvas has its own tabs: Health beside Summary in the run panel (counts, change since the previous run, findings by pack or by step) and beside Criteria in the step detail, which now holds acceptance criteria only. Arrow keys move between tabs. Review mode links to the step's Health tab. Plus a health badge on run lists and a health page per project with the trend over time. MCP: get_health and adopt_health_check.

• After a decision the step shows it: the approval panel opens with who approved or returned the step and when, the signature's meaning and time when signed, the note, the attached files and earlier decisions, and Change decision. Approvals are no longer tucked away as resolved comments. The step header and its card on the canvas show the decision too (with a mark when signed), and the run summary shows who accepted the run.

• Signature meanings come in the signer's language. Reviewed, Approved and Verified are translated; an organization adds a Czech label to its own meaning after a bar (QA release | Uvolnění QA). The signature keeps the label the signer saw, its id and the language. The signing dialog fills in the name from the profile, or from company sign-in.

• A signature now also covers the decision: its note and the SHA-256 of the attached files, pin and drawing. Earlier signatures keep verifying as they were made. The acceptance record lists the files attached to each step decision and the step's earlier decisions, and the evidence package holds the files under decisions/<step>/, listed in the signed manifest.

• Requirement references show on the canvas as chips: in the run info, under the step title, beside each criterion and in review mode. The index finds steps by them.

• Hosted runs and runs shared from CI tell you when health gets worse: a new error-level finding notifies the scenario's followers, the Slack and Teams channels that take "Health got worse" and webhooks with health.worse. Each schedule chooses errors (the default), warnings or no alerts, in Settings or with kaloko schedule add --health-alert.

• The agent skill checks health before handing work over and fixes what it can, without hiding findings.

Changed

• The semantic evaluator's prompt version is kaloko-jev-v1 (it was qawalk-jev-v1); answers cached before keep being used.

• Deleting a decision on a step goes into the audit log, and decisions on a run stay in the audit log as long as the run does (other entries still for two years).

• Check packs no longer decide acceptance. The checks of seo, a11y, perf, usability, console, errors and app still run on every walk, but they make up the run's health: shown apart on the canvas, counted by severity, compared with the previous run, and never part of a step's or run's verdict, the review queue, carried approvals or the acceptance record. A run no longer comes back red because of a meta description nobody asked for. The design pack still counts for acceptance. Runs shared earlier keep their verdicts.

• To keep a pack deciding as before, list it in the scenario: health: { acceptance: [seo] }.

0.14.0

Released 2026-10-07

Added

• An admin of a paid organization can start another organization under the same e-mail: Plan & billing has the link, and the sign-up page says the new one gets the same address. Free organizations and code trials without a card do not open the way, so they cannot be chained.

• --json is a contract now: stdout holds only the JSON answer, progress and notes go to stderr, and a failed command ends stderr with one line {"error":{"message":"…","code":"…","next":"…"}}. New on share (the run id and its URL), evaluate, build, validate, scenarios, auth list, browser status and ci preview-url.

• Two more exit codes: 3 when the Kaloko service cannot be reached (no connection at all), 4 when the token is missing, invalid, expired or revoked. 0, 1 and 2 mean what they meant; an error answer from the service other than 401 is still 1, and docs check, drift and design lint still report their findings with 1.

• Lists page through the API: runs, comments, notifications, tokens and the audit log take ?cursor= and name the next page in X-Next-Cursor and Link: <…>; rel="next". Bodies keep their shape. The MCP tools list_runs, list_notifications and list_comments take cursor and answer next.

• kaloko runs --all follows those pages to the last run instead of stopping at 300; --limit N lists at most N and --cursor goes on where a list stopped.

• kaloko export run|steps|docs|live|design as sub-commands, each with its own --help. The flag forms (export --live, --design, --format, --scenario) keep working.

• kaloko affected --fail-if-none: exit 1 when nothing is affected. --check does the same and says it has a new name; workflows written by kaloko ci use the new one.

• Durations take units: --timeout 90s, ci preview-url --wait 10m, share --expires 72h, prune --older-than 14d. A bare number means what it meant before.

• default_token_mode in a design environment names its default token mode; tokens_mode still works.

• A step picture in docs and changelogs plays the step's recording. When the accepted run has a recording (kaloko walk --record), the picture shows a play button; a click or tap plays it without sound, and the picture comes back at the end. Nothing plays by itself. ?media=video in a step reference says the step needs its recording, and kaloko docs check warns when the run has none; ?media=image keeps the picture only. It works in "Walk through it step by step" too.

• Recordings that published docs or a changelog play are kept as long as they play them, instead of expiring after 30 days, and they count in storage.

• Settings -> Security -> Experiments shows how Clef triage does in your organization: useful findings per kind, how well its probabilities hold, findings turned into comments, the time to the first review on runs with findings and without, and triage requests per useful finding. The same numbers come from GET …/findings/stats and MCP get_triage_report.

• Exports carry the recordings: Markdown, Docusaurus, VitePress and offline HTML get a player with the MP4 and WebM in assets/, MkDocs a picture linked to the recording; the PDF keeps the picture. MCP resolve_step_image and GET …/steps/resolve answer video with the MP4 and WebM addresses and the length.

• Required approvers are asked by name. When a run waits for them, each approver whose e-mail is in your Slack workspace gets a direct message with a link straight into review mode, and Slack and Teams channels that take the new event "Waiting for approval" name them (Teams workflows cannot write to one person). One reminder follows after 24 hours if nobody has decided; the hours, the reminder and quiet hours are in Settings -> Integrations, and admins can read and change them through GET and PATCH /api/v1/orgs/<org>/approval-messages. Slack connected earlier needs to be connected once more so Kaloko can look people up.

• Hosted runs start after a deploy. kaloko schedule hook <id> (or Settings -> Integrations -> Hosted runs -> Run after a deploy) makes a trigger URL and a secret, shown once. A POST with the secret as a bearer token, an

HMAC signature of the body (X-Kaloko-Signature) or Netlify's signed deploy notification starts the run at the next quarter hour. The hosted runs guide has copy-paste examples for curl, Vercel and Netlify. Business plan and up, at most 30 triggers an hour per schedule, each one in the audit log.

• An invitation that cannot be delivered says so. When the mail provider reports a hard bounce or a spam complaint, the pending invitation shows "Address bounced: check the spelling" in Settings -> People and in the Share dialog, the person who sent it gets an inbox item, and Kaloko stops sending it. GET /invitations names the mark in bounced.

• Links to the pricing page, the comparisons, the situations, the "for…" pages and every guide show their own card in Slack, LinkedIn, X and other previews: the page's title and description on the Kaloko card, in the page's language. The other pages keep the default card.

Changed

• New API tokens start with kal_. Tokens that start with qwk_ keep working until they expire.

• Domain verification shows the TXT record _kaloko.<domain> with kaloko-verify=…. A _qawalk record added earlier still verifies the domain.

• Names from before the rename print one line on stderr when used: the qawalk command, QAWALK_* variables, qawalk.config.yml and ~/.config/qawalk/.env. They keep working; kaloko doctor lists them with their new names.

• Public docs and changelogs take their language from ?lang=, as the app does. Links, the canonical address, hreflang, the sitemap and llms.txt use it; the former ?language= (docs) and ?locale= (changelog) redirect there, so old links keep working.

0.13.0

Released 2026-10-06

Added

• kaloko <command> --help (or -h) and kaloko help <command> show the help of that one command, with the options every command takes and the exit codes: 0 done, 1 checks found problems or the command failed, 2 a usage error. kaloko --version, -v and kaloko version print the version.

• kaloko start --json prints the run id and its folder, kaloko status --json what each step has captured.

• --org <slug> on every command works with another organization than org: in the config.

• API errors name their reason in code next to error. When the plan does not include something (HTTP 402) the answer links the organization's Plan & billing in billing_url, and the CLI prints that link.

• Webhooks carry Kaloko-Signature and Kaloko-Event next to QAwalk-Signature and QAwalk-Event, with the same values, so receivers written before the rename keep working.

• The trust page names a contact for reporting abuse (phishing, malware) on kaloko.app, files.kaloko.app and design previews at *.kaloko-usercontent.com.

• The trust page explains how Kaloko hides personal data in captures (masks, [hidden] in the stored HTML and the outline evaluators read, pii: redact) and links the how-to. The subprocessor list now names Workers AI at Cloudflare, which Clef triage uses only when an admin turns the experiment on.

• Every guide is also plain Markdown at its own address with .md added (/guides/ci.md, /cs/pruvodci/ci.md), for agents and answer engines. llms-full.txt now carries the pricing table and billing questions, the comparisons, the situations and the trust facts.

• The comparison pages answer the questions people ask about them.

Changed

• An option a command does not take gets a "Did you mean --env?" instead of being skipped quietly. prune, trash, restore, keep, archive, public, release and comment stop before they change anything; other commands say it and go on.

• A usage mistake (unknown command or option, missing argument) exits with code 2 and points to kaloko help

<command>, without the support line.

• runs, prune and screens take --env like the other commands; --environment still works. archive takes --off like keep and public (--undo still works), and prune takes --dry-run, which is what it does without --yes.

• A service account token on a plan without service accounts is answered with HTTP 402 (it was 403): the plan, not the token, has to change.

• Webhook requests say User-Agent: Kaloko-Webhooks/1.

• Pricing: the print-quality row is gone, a row shows that hiding personal data works on every plan, and another that the GitHub check on the pull request comes with Team. The site says plainly that your data stays readable on every plan and that exporting a run as a ZIP comes with Team.

• Terms of service, version 4: a trial started with an invitation code needs no card; if none is added, the organization returns to Free when the trial ends. The terms and the pricing page call the billing settings Plan & billing, as the app does.

• Kaloko's public changelog and help have one address per language, which search engines and link previews now understand. The Czech changelog page says that the release notes are written in English.

• /design and /cs/navrh lead straight to the guide on designing with your agent.

Fixed

• kaloko prune --env preview --yes pruned the runs of every environment: --env was not read. It now prunes only that environment.

• --key=value keeps everything after the first =: --map="Sign in=login" arrives whole.

• Commands run without org: in the config say how to set it instead of failing with a 404 on /orgs/undefined.

• kaloko runs says when the list stops at the newest 300 and how to narrow it.

• kaloko design import --json prints only the JSON on standard output; progress goes to standard error.

0.12.2

Released 2026-10-06

Fixed

• Czech changelog headings (Přidáno, Změněno, Opraveno, Odstraněno, Zastaralé, Bezpečnost) count as Added, Changed, Fixed and the other Keep a Changelog groups, so a Czech CHANGELOG.md is grouped like an English one.

• A form that needed a fresh second sign-in step (inviting someone, changing a setting) is sent again after the step instead of coming back empty. The fields wait in the browser tab, never on the service.

Changed

• Screens on the canvas cards stay sharp: thumbnails are 640 wide and scaled down step by step (one big jump made small text fall apart), and when you zoom in past what a thumbnail holds, the visible cards load the full screenshot. Runs shared before this get the zoom part too.

• Videos play on: once you play a step's video, the next step's video starts by itself in the step detail, and replay with video moves to the next step when a video ends (a decision waits for you, a still step shows its screenshot for three seconds). The next step's video loads while the current one plays.

• A step's video is at least 2.5 seconds long, held on its last frame, so a quick step no longer flashes past. A step where nothing moved keeps no video at all, and the canvas shows its screenshot.

• kaloko.com and www.kaloko.com lead to the same address on kaloko.app.

0.12.1

Released 2026-10-05

Added

• A trial code from us gives an organization's first subscription a longer free trial without a card. Open the sign-up link with the code, or enter it under Plan & billing. Two weeks before the trial ends, admins are asked for a card; without one the organization goes back to Free on its own and keeps its runs.

Changed

• kaloko share refuses unmasked personal data only in a run that hides it (pii on the run, scenario or environment, or mask: auto); other runs are shared with the same list as a warning. The site's own contacts no longer count: e-mails on the environment's domain, e-mails and phone numbers in the page footer, and those of an Organization or ContactPoint in JSON-LD (pii: { site_contacts: false } checks them again). Sample addresses such as you@company.com, placeholder text and SVG drawings are not read as personal data.

• Phone numbers written 603 123 456 count only next to a word like Telefon or Mobil, and amounts, EANs and order numbers no longer read as phone numbers. The nine-digit birth number needs its label (RČ), as does one without the slash.

• pii: pseudonymize gives one value the same stand-in in every run of the scenario, so screenshots and approvals carry over, and two values never share one. KALOKO_PII_SALT adds a secret of your own.

Fixed

• Hidden form fields, value attributes and personal data in URLs (also encoded, jan%40firma.cz) are masked.

• The capture record no longer keeps real data: page title, headings, JSON-LD, microdata, console lines, blocked requests, the URL and check evidence all go through the mask, and a capture whose masked page cannot be read stores nothing.

• Masking no longer changes what the page reacts to: console errors are read before it, data-* attributes are masked only in the stored HTML, and the keyboard walk runs on the real page with masked focus shots.

• Short masked values such as 2024 or Praha are no longer replaced across the whole run (width:100% stays).

• Recordings cover personal data in modal dialogs and popovers, open shadow DOM and same-origin frames, and on a new page from its first frame with everything masked so far. Screenshots of steps mask the same places.

• Values masked in one kaloko process are masked in the next process of the run too.

• Triage (experimental) fits long pages into Clef: two page tiles and smaller recording frames instead of four full tiles, and a request still too large is asked again with half the images. A Cloudflare token in the project that cannot use Workers AI hands triage to Kaloko instead of stopping it, and such a token is used only when evaluators.triage is set in kaloko.config.yml.

• A failure while masking leaves the page as it was.

• Long words without spaces no longer slow down detection.

0.12.0

Released 2026-10-05

Added

• Triage, an experiment an admin switches on in Settings -> Security -> Experiments. kaloko evaluate shows each screen, and a few frames of its recording, to Clef on Cloudflare, which names the kind of problem it sees: overlapping elements, cut-off text, a raw translation key, a placeholder, an error, an endless spinner, a layout that jumps. The canvas lists them under "Possible problems"; one click turns a right one into an ordinary comment, and findings never change a verdict. evaluators.triage in kaloko.config.yml tunes or turns it off.

• MCP tools get_findings and answer_finding.

• Masks hide personal data in everything a step stores, not only in the screenshot. The captured HTML and the page outline the evaluators read get [hidden] of the same length ([skryto] in Czech runs), and traces, recordings, crops, focus shots and the HTML view are covered too. Checks still read the real page, so their verdicts do not change.

• pii: redact on a scenario finds e-mails, phone numbers, birth numbers, IBANs, account numbers, dates of birth,

addresses and names in greetings without a selector, in Czech formats too; capture: { mask: auto } does it for one step. pii.allow in kaloko.config.yml lists values that only look personal, such as your support line.

• pii: pseudonymize replaces personal data with made-up values instead of boxes. One value gets the same stand-in for the whole run, so you can still see a name travel from the form to the summary and the e-mail.

• Captured e-mails hide their recipients (To and Cc, now captured too) whenever a step masks anything; greetings and payer blocks follow the step's masks.

• kaloko share checks the run before it uploads: an unmasked e-mail, phone number, birth number or IBAN stops it, with the step and the element. --allow-pii uploads anyway.

• An environment with pii: required gets no run that would keep personal data; kaloko start --pii redact (or pii: on the scenario or the environment) satisfies it.

• The canvas says "Personal data hidden" on such runs and outlines the masked places in the step detail, so a dark box is not taken for a bug in the app.

Changed

• The masks of a scenario and of its step now add up; before, a step's capture.mask replaced the scenario's.

• kaloko.app serves its own fonts and the canvas you open from disk carries them inside, so no page asks Google Fonts for anything and Google Fonts is off the subprocessor list on the trust page.

0.11.0

Released 2026-10-04

Added

• The approval page of kaloko login says where the request came from (city, country and address), which terminal asked and when, and starts with a plain warning: approve only a sign-in you started yourself, just now. When the request came from another country than the one you are in, the page says so in red.

• SCIM tokens are valid for a year. Settings -> Security -> SCIM shows the date, and "Renew for a year" keeps the same token, so nothing changes in your identity provider. Admins get a reminder in the inbox 30 days before a token expires. Tokens you already have run for a year from this release.

• A company sign-in over SAML set up before "Require a signed response" existed turns it on with one click in Settings -> Security. Kaloko offers the button once it has seen your identity provider sign its responses, so people keep signing in as before.

Changed

• A passkey used as the second step must ask for your PIN, fingerprint or face. A security key that signs without asking no longer counts; your account page names the key and says how to bring it back (set a PIN on it, or add another passkey).

• A run whose title, environment, branch, commit message or another field is too long is refused at the start of the upload with a message that names the field and the limit, instead of an unexpected error halfway through.

Fixed

• A browser that launched but then stopped answering no longer holds kaloko walk, evaluate --recheck or a capture until someone kills it. A new browser must open its first window within 20 seconds or it is closed and started once more, and later windows have the same limit. kaloko browser start, kaloko doctor, kaloko auth save, the PDF export and connecting to the shared browser now have the time limit and the second try too (KALOKO_LAUNCH_TIMEOUT, seconds).

• A js_equals check on a busy machine no longer fails when the page answers a little late: the expression keeps its 2-second limit, and Kaloko waits up to 17 seconds for the page to reply.

Security

• The sign-in cookie can be set only by kaloko.app itself. You stay signed in while it changes over. Your session also gets a new id at every sign-in and right after the second step.

• App and website pages tell the browser which features they never use (camera, microphone, location, payments and the like), and a Kaloko tab opened from another site is cut off from that site's window.

• Billing events from Stripe and pull request events from GitHub are applied once, also when someone sends a captured copy again.

0.10.0

Released 2026-10-04

Added

• A step can list its other states: states: [empty, error, loading]. Walks run the step script once per state (it gets state), a design draws cart/empty.html next to cart/index.html, and each state gets its own screenshots and verdicts. The canvas has a State switcher next to the language, and review mode and the grid show the state you pick. A criterion with states: [empty] is judged on the empty cart only.

• A design whose token set has several modes, such as two brands, is rendered in each of them. Switch modes on the canvas like colour schemes; tokens_mode in the environment stays the default. Light and dark alone still follow color_schemes. Variants of a stack step are captured in every state and mode too.

• kaloko capture --state, kaloko spec --state and a state option on the MCP step tools (get_step, get_step_spec, get_step_drift, get_step_assets, get_step_recording).

• A design step can come in variants of one template, such as an e-mail for each reason support can pick. Make it a stack step and give each variant a folder, design/<step>/<variant>/index.html. kaloko walk captures every variant in every viewport, kaloko sync sends them to the draft, and in a published version the step opens a list of its variants with their results. Each variant opens live. kaloko design lint reports a variant folder without its page, and both design exports include the variant folders.

• Web fonts on live design steps. An organization admin lists the addresses designs may take fonts, stylesheets and images from (Google Fonts, Adobe Fonts, your CDN) in Settings -> Security, and live steps load them instead of falling back to a system font. Scripts still load only from the design itself. List the addresses under origins of the design environment and kaloko design lint says which of them the organization does not allow yet.

• Prototypes remember things between steps: a cart filled on one step shows on the next, in Live mode and in the presentation. The state stays in the viewer's browser tab, every step opened alone still shows its own sample data, and "Start over" in the presentation clears it. Design agents use window.kaloko.state.

Changed

• Large runs are quicker on the canvas. On a run of 500 steps in seven languages on three devices, with the CPU slowed down to a phone's, a zoom step takes 33 ms instead of 133 ms, switching between desktop and mobile 650 ms instead of 910 ms, and the map is drawn once when it opens (it was drawn twice). Scrolling the step index no longer sorts every step on each frame.

• On kaloko.app a large run's canvas data and a scenario's history page come from a cache after the first visit (27 ms instead of 113 ms, 9 ms instead of 230 ms), and the changelog, the products page, the screen library and review mode ask the database far fewer times.

0.9.0

Released 2026-10-04

Added

• After a release, people who read the product see a short "What's new" bar in the app that links its illustrated changelog. It comes once per release and person; open the changelog or close the bar and it stays gone.

• The canvas tells you when the connection drops or Kaloko cannot save. Comments, approvals and verdicts wait and go out once you are back online, or with "Try again"; nothing is posted twice and nothing is lost quietly.

• kaloko doctor --fix installs a missing browser, creates missing folders and fills in org: from your token. An expired token points you to kaloko login.

• Long stack walks show roughly how much time is left.

Changed

• On a phone, review mode puts Approve and Return under your thumb at the bottom of the screen and says what you just decided and which step comes next.

• The first-steps list links straight to where each step is done: your newest run for comments and approvals, with the command to copy for the steps done in the terminal.

• Every error page has a way to write to support, with the address already in the e-mail, and a link to the status page. The app and website footers link the status page too.

• CLI errors say what happened and what to do next (sign in again, wait and retry, check the network) and end with support@sinfin.cz. Losing the connection no longer prints a stack trace.

Fixed

• A browser video encoder that stops responding no longer holds kaloko walk --record: the recording ends with a note and the walk goes on.

• Dark mode is easier to read: form fields have visible edges, the highlighted plan column on the pricing page is readable, and shortcut keys on the canvas's review buttons stand out.

0.8.0

Released 2026-10-04

Added

• Steps and scenarios say what kind of page they are: page: landing, work, list, settings, form or doc. The usability pack asks questions that fit: "one clear primary action" only on landing pages, and on lists, settings, forms and docs a question of their own. Without page, public pages count as landing pages and app screens as work pages, so a signed-in overview is no longer asked for a single call to action.

• A criterion can apply to some environments only: environments: [production] on hreflang pairs, a live demo or the production sitemap. On a local or staging walk it shows as not applicable to that environment and does not make the step partial.

• kaloko validate warns when a scenario walks several languages and a text check matches only one of them, for example pattern: "Plans and payment" without the Czech wording.

• Empty pages in the app say what to do next and give the command with a copy button: a product without a release or docs, its changelog and docs, an organization without runs, the runs and projects lists, the inbox and the screen library. People who only read a product see a plain sentence instead of a command.

• Kaloko's help covers the whole product, in English and Czech: two tutorials, how-to guides from writing a scenario to company sign-in, a reference of every CLI command, scenario key, check, pack, config key and MCP tool, and explanations of how verdicts are decided and how Kaloko maps to your applications.

• The website links Kaloko's own public changelog and help, so you can see both working before you set them up.



The home page · en · desktop

• Company sign-in over SAML can require a signed response: the identity provider must sign the whole response, not only the assertion. New connections start with it on. Connections set up earlier keep working as before, and Settings -> Security recommends turning it on.

Changed

• Error pages say why you got there and how to go on. A missing page offers the overview or another sign-in; a page that needs a role you do not have offers an e-mail to your organization's admins, already written; an expired link says to ask for a new one; an ended session signs you in and brings you back to the same address.

• Unsent text on the canvas survives. The verdict, return and issue notes keep their draft like comments do, and when your session expires while you write, the canvas says so, keeps the text and returns you to the same step after you sign in again.

Fixed

• A browser that starts but never answers no longer leaves kaloko walk, video or a check waiting: each launch has a time limit and one more try (KALOKO_LAUNCH_TIMEOUT in seconds, default 45).

• kaloko init proposes products from npm, Yarn and pnpm workspaces and from packages/ too, not only from apps/; shared libraries without a dev, start or serve script are left out.

• kaloko stability --pull names the button people use on kaloko.app: Mark a changing region.

• A table in published docs can hold a pipe inside a cell, written as \|.

• Hosted runs open only pages whose address leads to the public internet. An address that points into a private network is refused when you schedule the run and again before each run. Saved passwords and cookies are sent only to https addresses.

• Text and attribute checks in hosted runs finish quickly whatever the pattern. Patterns with backreferences or lookahead are reported as not evaluated there, with the reason; kaloko walk still checks them in full.

• Signing in, sign-up, invitation links and the second step have their own limits on repeated attempts. Exports and new API tokens have a generous hourly or daily limit per person.

0.7.1

Released 2026-10-04

Fixed

• kaloko walk of an Android, iOS, desktop or Electron app ends with the verdicts, like a walk of a website; before, the criteria stayed unchecked until you ran kaloko evaluate.

0.7.0

Released 2026-10-04

Added

• Products in the app. The main menu has Products: every product you read with its newest version, its docs, its modules and projects, who reads it and links to its changelog and docs. A project page names the products it belongs to. kaloko products sync puts the products of kaloko.config.yml on Kaloko before their first release, and kaloko products list shows which ones are there.



The Products page · en · desktop

• A trust page at kaloko.app/trust (in Czech at /cs/duvera): where your data lives, how it is encrypted, who can get in, which companies process it and where our SOC 2 preparation stands. Kaloko is not SOC 2 certified yet, and the page says so. The security contact is also in /.well-known/security.txt.



The trust page · en · desktop

• Seat warnings. When 80 % of the paid seats are taken, and again when all are, organization admins see a banner on the home page, in Settings and on the billing page, and get one e-mail per threshold in each billing period. GET /api/v1/orgs/<org>/usage reports the seats too.

• Yearly Enterprise subscriptions can go above their paid seats. Kaloko keeps the highest number of seats in each quarter; after the quarter, Kaloko billing reviews the true-up and invoices the extra seats for the rest of the subscription year or waives them. Seats added in the last quarter are not charged for that year: a week before the renewal the subscription rises to them and the renewal invoice includes them. The billing page shows the current quarter and every earlier true-up.

• Company sign-in over SAML accepts encrypted assertions. Create the organization's key pair under Settings -> Security -> Company sign-in; its certificate is in the SP metadata, and a new key pair keeps the old one working until you remove it.

• Single logout with your identity provider: signing out at the provider signs people out of Kaloko, and signing out of Kaloko can sign them out at the provider too.

• Sign-in from the app tile of your identity provider, off until an admin turns it on, with a default page people land

on.

Fixed

• The keyboard checks of kaloko walk no longer report a trap inside a third-party widget such as an anti-bot check, and judge a link wrapped over two lines and a small radio inside its label as people see them. Tab lists with arrow keys count as one tab stop.

• The canvas works better from the keyboard: every control shows a clear focus ring in light and dark mode, a "Skip to content" link jumps past the top bar, and while a step detail or another overlay is open, Tab stays in it instead of wandering through the hidden canvas. Screen readers get named side panels and dialogs.

• The canvas no longer jumps while it loads.

• A link from a comment notification opens the step with its comments in view, on a phone too.

• On a phone, a live page in the step detail fills the room above the sheet instead of a short strip, an open sheet leaves a strip of the preview that you can tap to lower it, and "Ctrl+Enter sends" shows only with a keyboard.

0.6.0

Released 2026-10-03

Added

• A visual changelog. kaloko changelog draft writes the Unreleased entry of CHANGELOG.md from the pull requests since the last release and adds before and after pictures of the screens your accepted runs show differently. kaloko release <version> records the release with its pictures, and kaloko.app shows it as a page you can share with clients, support or anyone outside development.

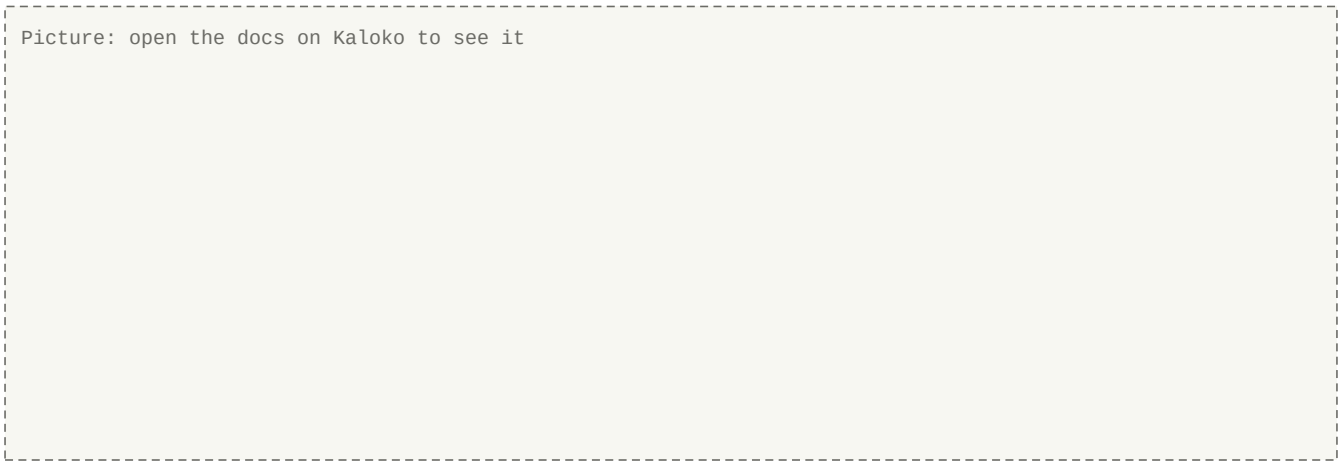
• Step references in Markdown: always shows the newest accepted screen, and ?version=2.4.0 the screen of that release.

• Products and modules in kaloko.config.yml for repositories with several apps; scenarios say which product they show (product:, module:). Without them the project is the product.

• Readers from your domain: everyone who signs in from a verified domain reads the changelogs, also of restricted projects (Business and up).

• kaloko init asks whether to keep a visual changelog, kaloko doctor checks that every product's version can be read, and agents get get_changelog, draft_changelog_entry and resolve_step_image over MCP.

• Docs that keep up with the app. Guides in your docs/ folder show steps instead of screenshots; kaloko docs publish puts them on kaloko.app with a version picker, language and screen switches and search, and every picture opens its step on the canvas.



A guide with its pictures · en · desktop

• When a screen changes after its guide was written, the section says "the screen changed in 2.4 — check the text", and kaloko docs check lists it with broken references, missing translations and changelog items without a picture, so CI can stop on them.

• A guide plays as a walkthrough, one step at a time with its words as the caption, and readers tick off the steps they tried.

• Exports for your own docs site: kaloko export --format md, docusaurus, vitepress, mkdocs, html, pdf or json, with the pictures as files. The docs and changelog pages offer the same downloads.

• A "What's new" e-mail: readers ask for it on a product's changelog and get each new release with its pictures in their language (Business and up).

• kaloko ci github --release records every release that release-please, changesets or semantic-release publishes.

• Issues in Jira and Linear from a returned step or a failing criterion, on the canvas, in review mode or with kaloko issue create. The issue gets the screenshot, the criterion, the notes and a link back, and the comment on the step is resolved when the issue is done. Branches and pull request titles that name an issue link the run to it.

• Slack and Teams channels per project, each with the events it wants and quiet hours. Restricted projects reach only channels chosen for them.

• A short list of first steps on the home page (share, comment, approve, invite, CI, release) until the team has done them.

• Hosted runs: Kaloko opens the pages of a read-only scenario every day or week and shares the run (kaloko schedule add, Business). kaloko schedule export github writes a scheduled workflow for the full walk.

0.5.2

Released 2026-10-03

Added

• The first version of the visual changelog: kaloko changelog draft, kaloko release, step references and the changelog page on kaloko.app with a before and after slider. It is described in full under 0.6.0.

0.5.1

Released 2026-10-03

Added

• Swipe and onion skin compare next to side by side and diff, in the step detail and in review mode.

• Regions to ignore can be drawn by hand on a capture. The canvas diff skips them, and kaloko stability --pull writes them into the scenario.

• A grid (key G) shows one step in every language and screen size at once, also on a phone and in review mode.

• Comments can carry drawn boxes, arrows and lines, and attached images. Agents get the marked region in pixels.

• An accepted run has a signed acceptance record: a PDF with thumbnails and a printable page. kaloko signoff --verify checks the signature.

0.5.0

Released 2026-10-03

Added

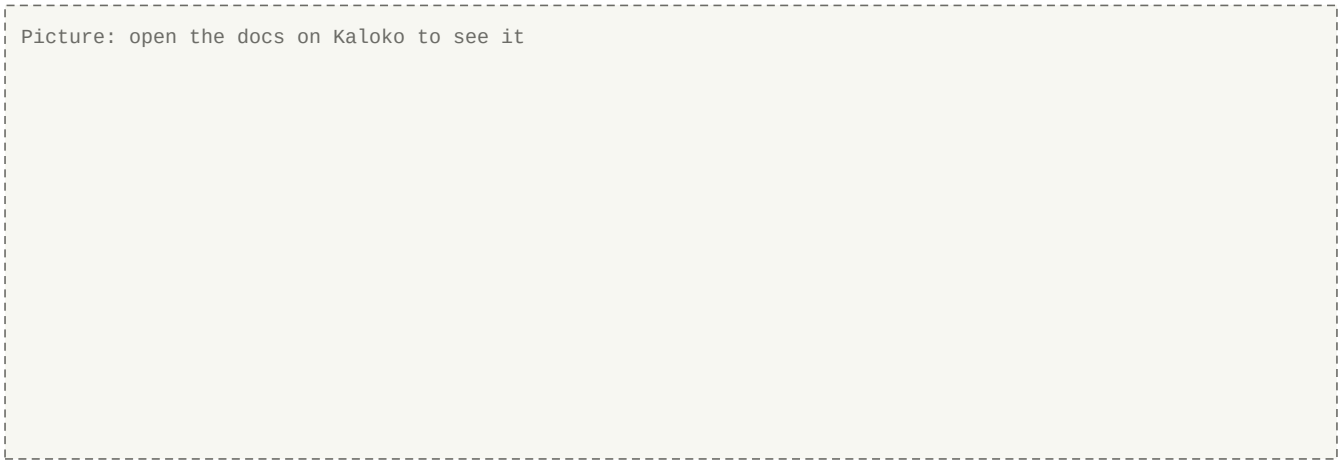
• Checks of the page itself: the address a step ends on, computed styles of an element and values the page exposes, so scenarios no longer need data-qa markers.

• Runs in WebKit and Firefox besides Chromium, on device presets (phones, foldables, tablets, TV), in dark mode, with reduced motion, forced colours or as an installed web app. Every step is captured in each combination.

• App walks on several devices at once: Android emulators, iOS simulators, a real iPhone and Android

WebViews, and Windows apps next to macOS ones.

• One picker on the canvas for language, browser and colour scheme that stays usable with many values.



The canvas of a run · en · desktop

• Workflows written by kaloko ci install the browser engines your scenarios walk.

Fixed

• Design drafts keep the browsers, schemes and device presets of their scope.

• An e-mail captured during a walk in several browsers counts once.

• Language and theme switchers work in the phone menu.

• Ignored regions, imported screenshots and cached evaluator answers carry over between capture variants.

0.4.11

Released 2026-10-03

Added

• Recordings: kaloko walk --record films each step from the previous screen to its capture, or the whole flow with a chapter per step, and keeps a trace of actions, requests and console messages for every step.

• The canvas plays a step's recording and shows its trace.



A recording on the canvas · en · desktop

• kaloko video export turns the recordings into one walkthrough video with a title card per step.

• Shared runs carry their recordings on the Business and Enterprise plans, kept for 30 days unless the organization chooses otherwise.

Fixed

• A recording too large to upload stays in the local preview instead of failing the share.

0.4.10

Released 2026-10-03

Added

• kaloko ci github|gitlab writes a workflow that walks what each pull request changes on its preview deployment and comments on the pull request; kaloko ci preview-url finds the preview address in CI.

• kaloko import lays out screenshots, recordings and traces that an agent or Playwright already produced as a run.

• kaloko walk --affected and kaloko affected walk only the steps that the changed files reach, and say why.

• kaloko walk --heal proposes a new selector when the page lost the old one, and applies it once the step passes.

• Several languages walk at the same time (--concurrency).

• kaloko init recognises the framework, the dev server, languages and main routes, and drafts a first scenario.

• Uploads skip screenshots the service already has and resume after a broken connection.

0.4.9

Released 2026-10-03

Added

• A vision evaluator that looks at the screenshot and says why, keyboard and screen reader checks, and regions that change on every run can be ignored.

• Dev mode in the step detail of a design: box model, tokens, redlines, states and assets.



Dev mode in the step detail · en · desktop

• kaloko spec tells an agent what to build for an approved design step, and kaloko drift what its implementation does differently.

Fixed

• The canvas on phones: a smaller stack badge, readable stack lists, the step header above its tabs and no sideways scrolling.



A stack page on a phone · en · mobile

• Page counts in Czech use the right plural.

0.4.8

Released 2026-10-03

Added

• Review mode: the canvas walks a person through what still needs a decision, with keyboard shortcuts, and kaloko review lists the same for agents.

• Approvals carry over to steps whose screenshots did not change since the last accepted run.

• Every new organization gets a sample run with a guided first review.

• Two-step verification with passkeys or an authenticator app, and an organization rule that requires it.

• An audit log for admins, tokens limited to chosen projects, and company sign-in with SAML or OIDC, SCIM and an IP allowlist on Enterprise.

Fixed

• Plan badges on the pricing page no longer overflow on phones.

0.4.7

Released 2026-10-03

Added

• Sign in with Google, Microsoft or GitHub where the service offers it.



The sign-in page · en · desktop

• kaloko login approves a token in the browser instead of copying it from Settings.

• Invitations: people outside the organization join single projects as guests, and the Share dialog of a project shows who has access.

0.4.5

Released 2026-10-03

Added

• A Designer role and API tokens with chosen scopes (read, comment, design, upload).

• Focus mode shows the preview alone, and the step detail has a layout for phones.

0.4.4

Released 2026-10-02

Fixed

• kaloko evaluate --recheck says that it asks the semantic questions again too.

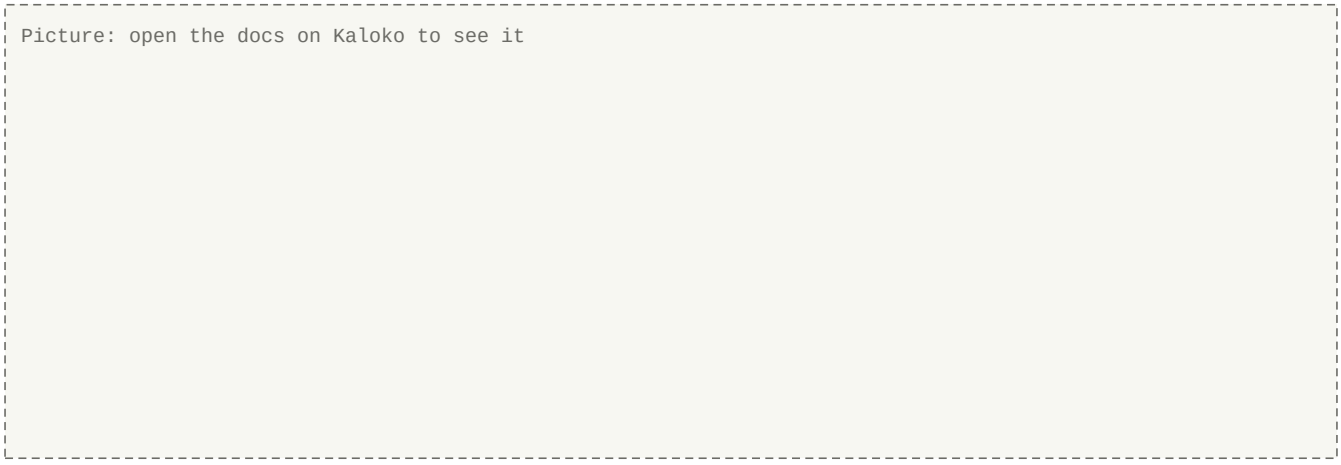
• A lane on the canvas is as tall as its tallest card.

0.4.3

Released 2026-10-02

Fixed

• Signed-in pages that Kaloko's own walk found wanting: labelled fields, real settings tabs, cards on phones and the whole inbox.



Settings · en · desktop

• The canvas has a main landmark, readable badges and a visible compare button, and no longer shifts while loading.

• Pages load their fonts sooner and keep their layout when the font arrives.

0.4.2

Released 2026-10-02

Added

• The Kaloko wordmark with a yellow marker and a new icon.



The home page · en · desktop

• The Czech website lives under /cs/, so each language has its own address.

• Pages of a stack may carry their own purpose for the semantic evaluator.

• Live design revisions open on their own domain.

Fixed

• kaloko doctor prints the config file it actually read.

0.4.1

Released 2026-10-02

Added

• links_ok and hreflang_pairs checks for pages and stacks.

Fixed

• Unknown addresses on kaloko.app answer 404 instead of the sign-in form.

• Every language version of the website answers at its own address.

0.4.0

Released 2026-10-02

Changed

• QAwalk is now Kaloko: the CLI is kaloko (the qawalk command and qawalk.config.yml keep working) and the service lives at kaloko.app.



The landing page · en · desktop

Added

• Kaloko Design presents interactive HTML designs, lints pages that must work from disk, and exports a design in open formats.

0.3.0

Released 2026-10-02

Changed

• The CLI is on npm as kaloko instead of @qawalk/cli. The qawalk command, qawalk.config.yml, the QAWALK_* variables and ~/.config/qawalk/.env keep working, and kaloko init replaces the old skill.

• The app and the website share kaloko.app. Pages on qawalk.com redirect there; the API, MCP, webhooks and file links on the old addresses keep answering.

Added

• A "Group by issue" switch on the lists of runs and scenarios. The lists remember it, and "Latest per scenario" too.

0.2.22

Released 2026-09-30

Fixed

• Large runs open faster on the canvas, and kaloko evaluate needs less memory for big stacks.

0.2.21

Released 2026-09-30

Fixed

• A kept run cannot be trashed, and releasing it gives it at least a week. Kept design versions are not pruned.

• The "no project" filter on runs works, failed step details load again, and an unusual cookie no longer breaks the lists.

• Lists, the home page and scenario history load faster.

0.2.20

Released 2026-09-30

Added

• The project you pick is remembered across the lists and the scenario browser on the canvas.

• Scenarios and runs of one issue, or of one pull request when there is no issue, are listed together.

0.2.19

Released 2026-09-30

Changed

• Shortcuts and help open from a keyboard button in the top bar, and the list now includes <- -> and Esc.

0.2.18

Released 2026-09-29

Changed

• A shorter top bar on the canvas: Projects, Scenarios and Runs, the view switch, and the inbox at the right end.

0.2.17

Released 2026-09-29

Changed

• Projects, scenarios and runs share one layout with the navigation on the left, so switching views does not jump. "All runs" is now "Runs".

• The run verdict has Accept and Return as the main buttons; note, baseline and public sit below them.

0.2.16

Released 2026-09-29

Added

• A projects page: scenarios, runs and how the newest runs stand in each project.

• Console errors in the step detail are highlighted, wrapped and can be copied one by one or all at once.

• Times follow the reader's time zone, set in Settings -> Profile or taken from the browser.

Changed

• Screen size switches say the name and width (Desktop 1440, Mobile 360), and a click anywhere on a card opens the step.

0.2.15

Released 2026-09-29

Changed

• The step detail has three tabs: Criteria, Page and Technical. Criteria that need attention come first; passing ones take one line.

• When only phones are shown, the map shows larger phone screenshots.

0.2.14

Released 2026-09-29

Changed

• Evidence of built-in checks reads as a sentence in Czech or English ("TTFB 950 ms, the limit is 800 ms"); the raw text stays under Technical detail.

0.2.13

Released 2026-09-29

Added

• Every criterion of the built-in packs says why it matters and how to fix a failure, in Czech and English. The walk summary prints the fix for the agent.

0.2.12

Released 2026-09-29

Changed

• The run summary sorts problems by what to do: what failed and why, what a person should decide, and screens that look off.

0.2.11

Released 2026-09-29

Added

• Quick signals on every screen: the evaluator also says whether the page does what it is for, is in the expected language or looks broken. Signals never change a verdict.

• kaloko walk evaluates while it walks and ends with a list of what failed or is unsure.

0.2.10

Released 2026-09-29

Added

• Every criterion says how it was checked, in words: the check as a sentence, or the question the AI was asked with its score and the pass threshold.

• Hosted AI evaluation from the Team plan up, with a monthly allowance shown on the billing page and in kaloko doctor. Your own evaluator key still works.

• Design tokens in the current DTCG format, with aliases, modes and deprecated tokens; kaloko tokens validate.

Fixed

• Manual decisions show the criterion on its own line, with reason and verdict below.

0.2.9

Released 2026-09-29

Added

• Scenarios come first in the navigation, and Runs show the newest run per scenario unless you filter.

• Keep a run so it never expires, prune old runs (dry run first) and archive scenarios you no longer walk, in the app or with kaloko runs, trash, restore, prune, keep and archive.

0.2.8

Released 2026-09-29

Changed

• A claimed sign-in domain lets people join only once it is verified.

• Comments on public runs name nobody, and upload tokens cannot delete runs.

• Below 1500 px the navigation and view options fold into one menu.

Fixed

• The canvas shows the map sooner and loads step details after it; large runs upload faster.

• When the service asks the CLI to slow down, it waits and tries again.

0.2.7

Released 2026-09-29

Added

• Comments on steps, whole runs and whole flows: threads, pins on the screenshot, resolve and reopen. Mentions are mailed at once, the rest goes to the digest, and agents use kaloko comment or MCP.

• Kaloko Design: design drafts as live HTML steps on the canvas, versions, an inspector with token names, comments pinned to elements, edits on the canvas, branches linked to git, and a design pack that checks an implementation against the approved design.

• An HTML view of captured pages (experimental) that can be forked into a design draft.

Changed

• A calmer run view on smaller screens: one top bar, view options in a popover, help behind ?.

Fixed

• A long step path no longer widens its card over the next one.

• macOS desktop captures ask for the Screen Recording and Accessibility permissions they need.

0.2.6

Released 2026-09-29

Added

• Mobile and desktop apps: Android, iOS Simulator, Electron and macOS windows, with UI tree checks, touch target sizes, crash detection and an app pack. kaloko capture --image imports screens from other tools.

• A run index next to the map (key I): every step in a list with filters, search and CSV, and stacks as folders.

0.2.5

Released 2026-09-29

Added

• The HTTP status of every capture, an errors pack for error pages, and {random} in a path for an address that cannot exist. kaloko draft adds a not-found step.

• The run card links the issue next to the pull request.

0.2.4

Released 2026-09-29

Added

• Stacks: one step stands for many pages of one template, taken from a list, a sitemap, a crawl or a script. The canvas shows them as a stack with an index you can filter, search, sort and export as CSV.



A stack opened as a folder · en · desktop

0.2.3

Released 2026-09-28

Added

• acceptDialogs() for step scripts, flags on text checks, kaloko evaluate --recheck after a scenario fix, and kaloko

walk --ephemeral for parallel walks.

Fixed

• kaloko mail reads only messages received since the run started.

• A walk captures fresh screens after --steps, basic auth survives a redirect, and only one walk at a time uses a shared browser.

0.2.2

Released 2026-09-28

Added

• kaloko draft <url…> writes a first scenario and acceptance plan for public pages.

• Check packs a11y, perf and console.

• Live presence on the canvas: who is looking, their cursors, and following someone's view.

• Large flows stay readable: lanes with headers, long lanes wrapped into rows, tidy edges and a zoomed-out view.

• Scenario history with pass rate and criterion stability, and kaloko calibrate to tune evaluator thresholds.

• Print-quality captures for documentation, with masks over personal data and numbered marks.

• The newest capture of every step at a stable address, a list of changes, signed webhooks and kaloko export --scenario for documentation (Business and up).

• GitHub checks on pull requests that complete when the run is accepted or returned.

• A sign-in domain can be proved by signing in with Google Workspace or Microsoft 365, and Settings shows the DNS host with copyable record fields.

Fixed

• The overview follows reviewers' criterion verdicts.

• Public mailbox domains such as gmail.com cannot be claimed.

• A walk stops a session after three steps fail with the same error, instead of trying a wrong password again and again.

• An open menu or popover stays on the capture, signed links in the page lose their credentials, and text check evidence shows the actual match.

• kaloko share says when the plan grants a shorter lifetime than requested.

0.2.1

Released 2026-09-25

Added

• Protected environments: HTTP Basic, service-token headers and client certificates, sent only to the environment's own addresses.

• Accounts per scenario with password and TOTP, and kaloko auth save for a single sign-on a person does once.

• ${VAR} in base_url for review apps, and secrets named in the config are removed from captures.

• Any mailbox through a script adapter, and kaloko mail --file for saved messages.

0.2.0

Released 2026-09-25

Added

• The CLI is on npm (as @qawalk/cli until 0.3.0).

• Plans with a billing page, a 14-day trial and a public pricing page.

• Named sessions: several people in one flow, each in their own browser, shown in swimlanes on the canvas.

• Run verdicts, a baseline run, required approvers and reviewer verdicts for manual criteria.

• Compare a run with the baseline or any earlier run: pixel differences, changed criteria and evidence, and kaloko compare.

• Replay walks the run one step at a time with a branch choice at decisions, and the canvas has a minimap.

• An inbox with notifications, a daily or weekly digest, and Slack.

• kaloko share --pr comments on the pull request; kaloko export and the canvas download a run as a ZIP.

• An MCP server for agents.

• Check packs seo and usability, plan coverage with kaloko validate --coverage, and flaky criteria.

• A public demo run anyone can open without signing in.

• Projects per scenario, and a runs list filtered by project, scenario, verdict, author, environment and branch.

• Sign-in domains verified by a DNS record, token scopes and expiry, a list of your sessions, and a 7-day trash.

• kaloko doctor checks your setup and says what to fix.

• Guides on the website for use cases and agents.

0.1.0

Released 2026-09-23

Added

• The first version. The CLI walks scenarios in Chrome, captures every step, runs the checks and asks an AI evaluator what a check cannot answer. A read-only policy keeps production safe.

• The canvas shows the run as a map of screens with criteria, page metadata (Open Graph, hreflang, structured data, headings) and links to a single step. A click on a criterion highlights it on the screenshot.

• The agent skill for Claude Code and Codex, installed with kaloko init --agent.

• The service: organizations by e-mail domain, sign-in by link, roles and API tokens. kaloko share uploads a run, reviewers approve or comment on each step, and kaloko feedback brings their notes back to the agent.

• The app and the website in Czech and English, with sign-up for new organizations.