

The 40-point SaaS UX audit checklist.

The exact list I run on every paid audit, grouped by the eight places SaaS products leak users. Open your product in a fresh account, work through the groups in order, and mark each check pass, fail, or fail with evidence.

40 CHECKS	8 GROUPS	~2h TO RUN	4 SEVERITY LEVELS
---------------------	--------------------	----------------------	-----------------------------

Severity scale

Blocker. Users cannot complete the task.

Major. Most users struggle or abandon.

Minor. It slows people down but they get through.

Polish. Consistency and craft.

Add an effort estimate, S, M or L, to every failure so your team can pick the high-impact, low-effort fixes first.

01 Onboarding and first run

Where most of the churn happens, and where fixes pay back fastest.

- ☐ **Time to first value.** How many screens and decisions sit between signup and the moment the product does something useful. More than three is a finding.
- ☐ **Empty states.** Every list, board and dashboard with no data yet needs one obvious next action. A blank screen is not a state, it is a dead end.
- ☐ **Setup questions.** Every field in the onboarding form should be justified by something the product does with it today. Anything else is delayed.
- ☐ **Progress and escape.** Users can see how far along they are and can skip to the product. Forced tours get abandoned.
- ☐ **The first success.** Something visible confirms the user got the thing they came for. If the product is quiet at that moment, the moment is lost.

02 Navigation and information architecture

Does the structure match the way users think about their work?

- ☐ **The home screen.** It should show the object users think in: the deal, the file, the ticket. Dashboards are for managers, and managers are rarely the daily user.
- ☐ **Depth of key actions.** The five most common actions are reachable in one click from wherever the user is. I count clicks for each.
- ☐ **Label language.** Navigation labels use the words customers use in support tickets, not internal feature names.
- ☐ **Wayfinding.** Users can tell where they are, how they got there, and how to get back. Breadcrumbs, active states and page titles all agree.
- ☐ **Search.** If the product has more than one type of object, search exists, finds all of them, and shows what type each result is.

03 The core workflow

The one task the product is for. This section usually produces the highest-severity findings.

- ☐ **Interaction cost.** The most repeated action has the lowest possible cost: fewest clicks, no confirmation dialog unless the action is destructive, keyboard support.
- ☐ **Reversibility.** Users can undo. Where undo is impossible, the confirmation explains what will happen, not just asks "Are you sure?".
- ☐ **State visibility.** Saving, syncing, processing and failure states are visible where the action happened, not in a corner toast that disappears.
- ☐ **Bulk actions.** Anything users do to one item, they will eventually need to do to fifty. Multi-select and bulk edit exist or are on the roadmap.
- ☐ **Defaults.** New objects start with sensible defaults so the happy path is "create and continue", not "create and configure".

04 Forms and data entry

Forms are where users type, wait and give up. Small fixes, large effect.

- ☐ **Field count.** Every field on every form earns its place. Optional fields are marked, or removed.
- ☐ **Inline validation.** Errors appear next to the field, in plain language, at the moment the user can fix them. No red banner at the top listing three problems.
- ☐ **Input types.** Email, phone, number and date fields use the right keyboard and formatting. Dates especially.
- ☐ **Autosave and recovery.** Long forms save drafts. Losing ten minutes of typing to a session timeout is a Blocker.
- ☐ **Button labels.** Primary buttons say what happens: "Send invoice", not "Submit". Secondary actions look secondary.

05 Feedback, errors and system status

How the product talks back when things go right or wrong.

- ☐ **Error copy.** Errors say what happened and what to do next. Error codes and stack traces never reach the user.
- ☐ **Loading.** Anything slower than a second shows progress. Anything slower than five explains why and lets the user do something else.
- ☐ **Success confirmation.** Completed actions are confirmed in context, and the confirmation links to the result.
- ☐ **Notifications.** In-app notifications are grouped, dismissable and actionable. A bell with a red badge that opens a wall of text is a finding.
- ☐ **Offline and failure.** When a request fails the user keeps their input and can retry. Silent failure is a Blocker.

06 Visual consistency and design system health

Inconsistency is not a taste problem. It makes the product harder to learn.

- ☐ **Component drift.** I count the distinct button styles, input styles and modal patterns. More than one of each is drift, and drift means the team has no design system in practice.
- ☐ **Type hierarchy.** Titles, labels and body text follow a small, consistent scale. Users should never wonder which text is the heading.
- ☐ **Spacing rhythm.** Cards, lists and forms share a spacing scale. Uneven spacing reads as "unfinished" even when users cannot say why.
- ☐ **Color meaning.** Green, red, yellow and the accent color each mean one thing across the product. Red for both "delete" and "selected" is a finding.
- ☐ **Iconography.** Icons come from one set, carry labels where the meaning is not universal, and are not the only way to identify an action.

07 **Billing, plans and account**

The screens that decide revenue and get the least design attention.

- ☐ **Plan comparison.** Plans are comparable on one screen, in the language of outcomes, with the recommended plan visible but not forced.
- ☐ **Upgrade moment.** Limits are shown before they are hit, and the upgrade path is one click from the limit, not buried in settings.
- ☐ **Trial ending.** Users know what happens when the trial ends, what they will lose, and how to keep it. No surprises on day fourteen.
- ☐ **Invoices and receipts.** Finance people can find, download and forward invoices without asking support. This is a top-five support ticket in most SaaS.
- ☐ **Cancel and downgrade.** Cancellation is possible without a call. Retention offers are fine, dark patterns are a Major finding and a churn multiplier.

08 **Accessibility and performance**

Not a compliance box. These are the findings that affect everyone on a bad day, on a small screen, or on a slow connection.

- ☐ **Contrast and text size.** Body text meets contrast guidelines and is at least 14px in the interface, 16px in forms. Grey-on-grey placeholder text is a repeat offender.
- ☐ **Keyboard and focus.** Every action is reachable by keyboard, focus is visible, and modals trap focus then return it.
- ☐ **Responsive breakpoints.** The core workflow works on a laptop at 1280px and on a phone. Data tables in particular need a plan for small screens.
- ☐ **Perceived speed.** The most used screens render something useful within a second. Skeletons beat spinners, cached data beats both.
- ☐ **Session and auth.** Sessions last long enough for real work, re-authentication keeps the user where they were, and magic links or SSO are there for teams that need them.

Fix list

Write every failure with evidence here, then sort by severity and effort. The top five are the ones to ship first.

#	FINDING	EVIDENCE	SEVERITY	EFFORT
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Want the three fixes that matter most?

Book a free 30-minute product review at veljanoski.com/book-a-call. I open your product live, run this list with you, and tell you what I would fix first.