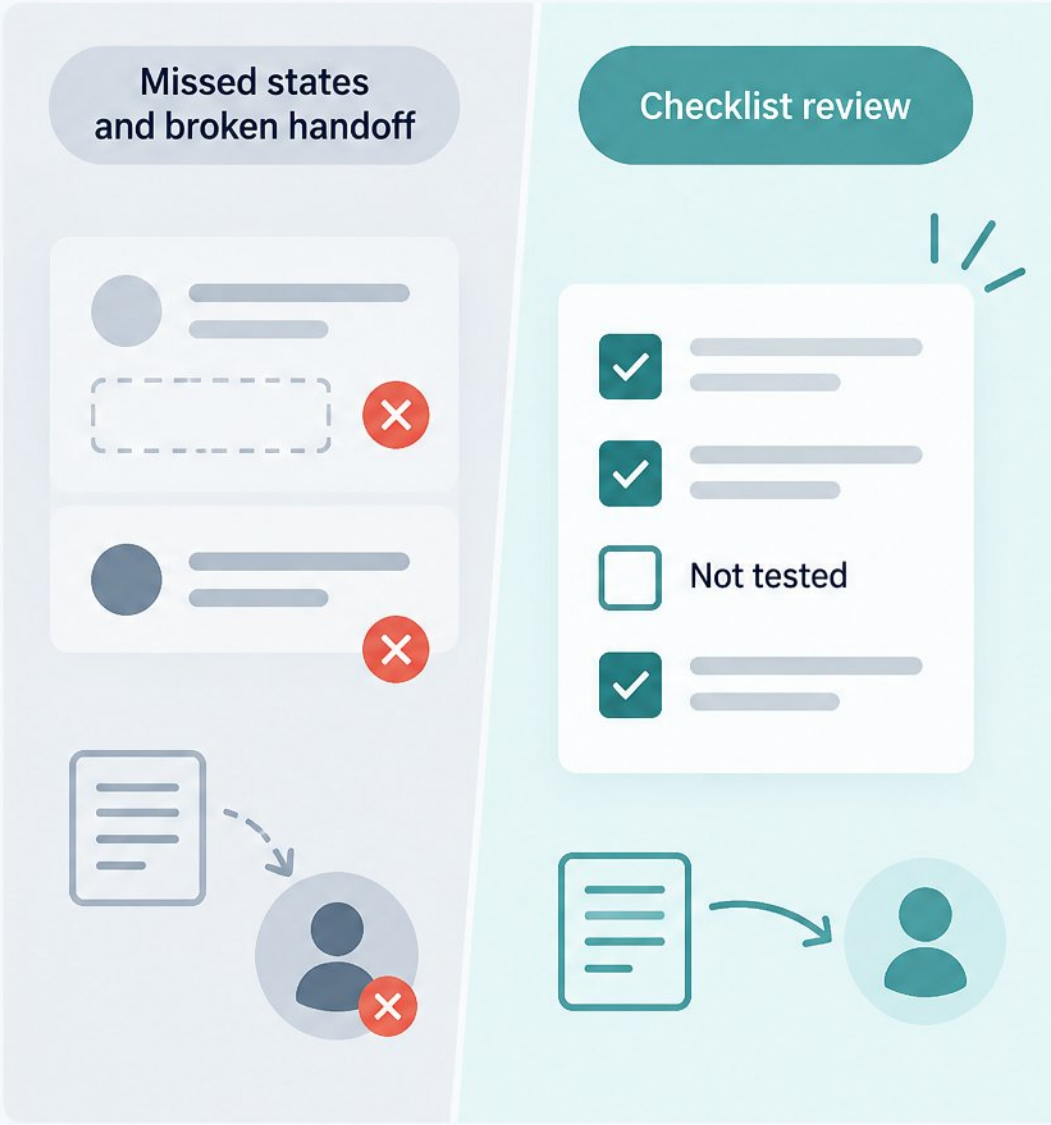
Creating a UX Design Checklist

- Checklists reduce oversight, improve consistency, and support cross-team handoffs
- A UX checklist is a review aid — not a rigid process or research substitute
- Four uses: design review, QA/acceptance, onboarding, and teaching
- Roadmap: principles, architecture, item design, review workflows, measurement, governance
- Built for designers, product teams, reviewers, educators, and content creators

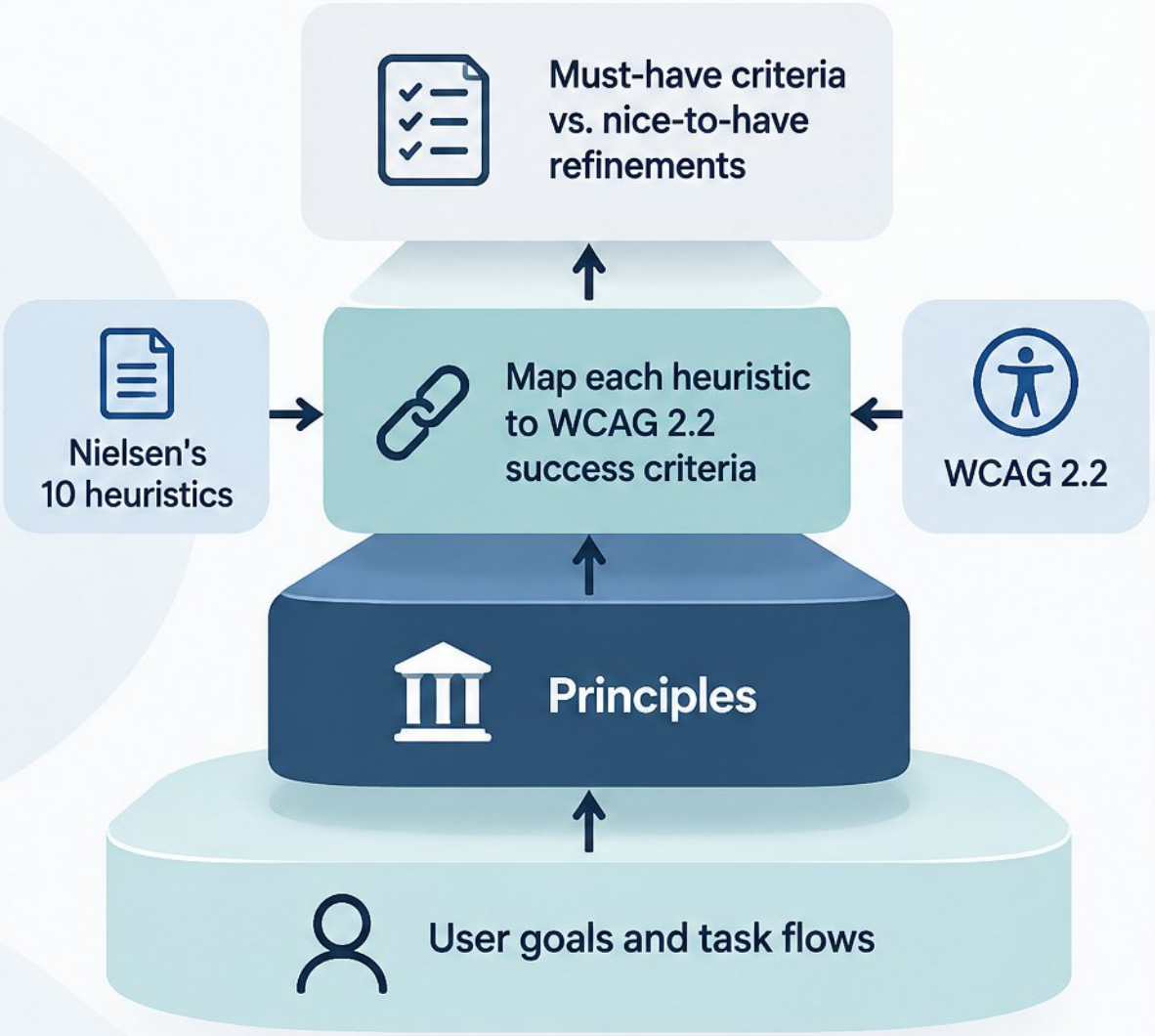


What Checklists Solve — and What They Don't

- Catch missed accessibility states, inconsistent patterns, incomplete edge cases
- Distinguish from heuristic evaluations, design-system guidelines, and QA scripts
- Agoda: reviews against written principles raised issue-free handoffs 50% (pilot), 171% (beta)
- 81% of reviews completed within one hour, with no hit to development velocity
- Checklists standardize the questions asked — not judgment, not user research
- 'Not tested' is a distinct state; gaps must never be mistaken for passes



Principles Before Items



- Start from user goals and task flows, not UI components
- Anchor to Nielsen's 10 heuristics without duplicating them wholesale
- Map each heuristic to WCAG 2.2 success criteria, not two parallel lists
- WCAG 2.2: 4 principles, 13 guidelines, testable criteria at A/AA/AAA
- Separate must-have criteria from nice-to-have refinements; define severity early
- Favor observable, testable criteria; fit the checklist to its audience

Anatomy of a Well-Formed Checklist Item

- Required fields: category, criterion, rationale, evidence/verification method, severity, owner
- Write testable statements with clear pass/fail or rating criteria
- Strong: “Primary action is visually distinct at all breakpoints” — Weak: “Design looks good”
- Split compound items: one check per item, no bundling
- Attach evidence: screenshots, prototype links, a11y scan results, focus-order annotations
- Rationale and evidence fields sharpen reviewer calibration and reduce taste calls

A well-formed checklist item

☐ Primary action is visually distinct at all breakpoints

Category	Visual design
Criterion	Primary action is visually distinct at all breakpoints
Rationale	Helps users quickly identify the primary action across different screen sizes.
Evidence/verification method	Screenshots, prototype links, a11y scan results, focus-order annotations
Severity	Major
Owner	Design team

All required fields, clearly defined

A weak checklist item

☐ Design looks good

Vague and not testable

Coverage Map: What Belongs in a UX Checklist

From research to impact — a complete view of what to check.

Research and discovery

problem framing, user evidence, competitive context

Understand

IA and content

labeling, hierarchy, microcopy, plain language at ~8th-grade level

Structure

Interaction and flows

navigation, forms, plus empty, loading, error, success, timeout states

Guide

Visual and system consistency

design tokens, components, spacing scale, responsive behavior

Unify

Accessibility

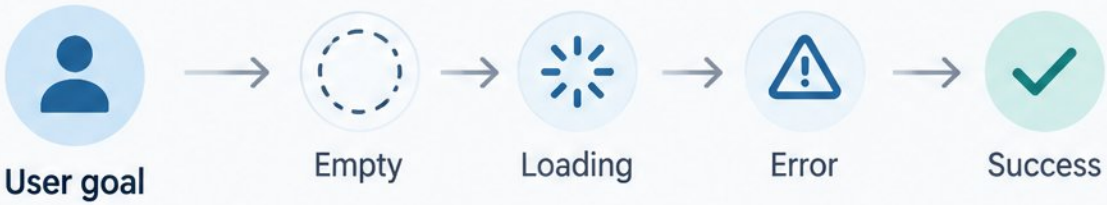
WCAG 2.2, keyboard and screen-reader paths, 4.5:1 text contrast, 3:1 UI contrast

Include

Performance

Core Web Vitals as design checks — LCP < 2.5s, INP < 200ms, CLS < 0.1

Keep it fast



Scope to maturity

MVP vs. mature product; pre-launch vs. continuous audit

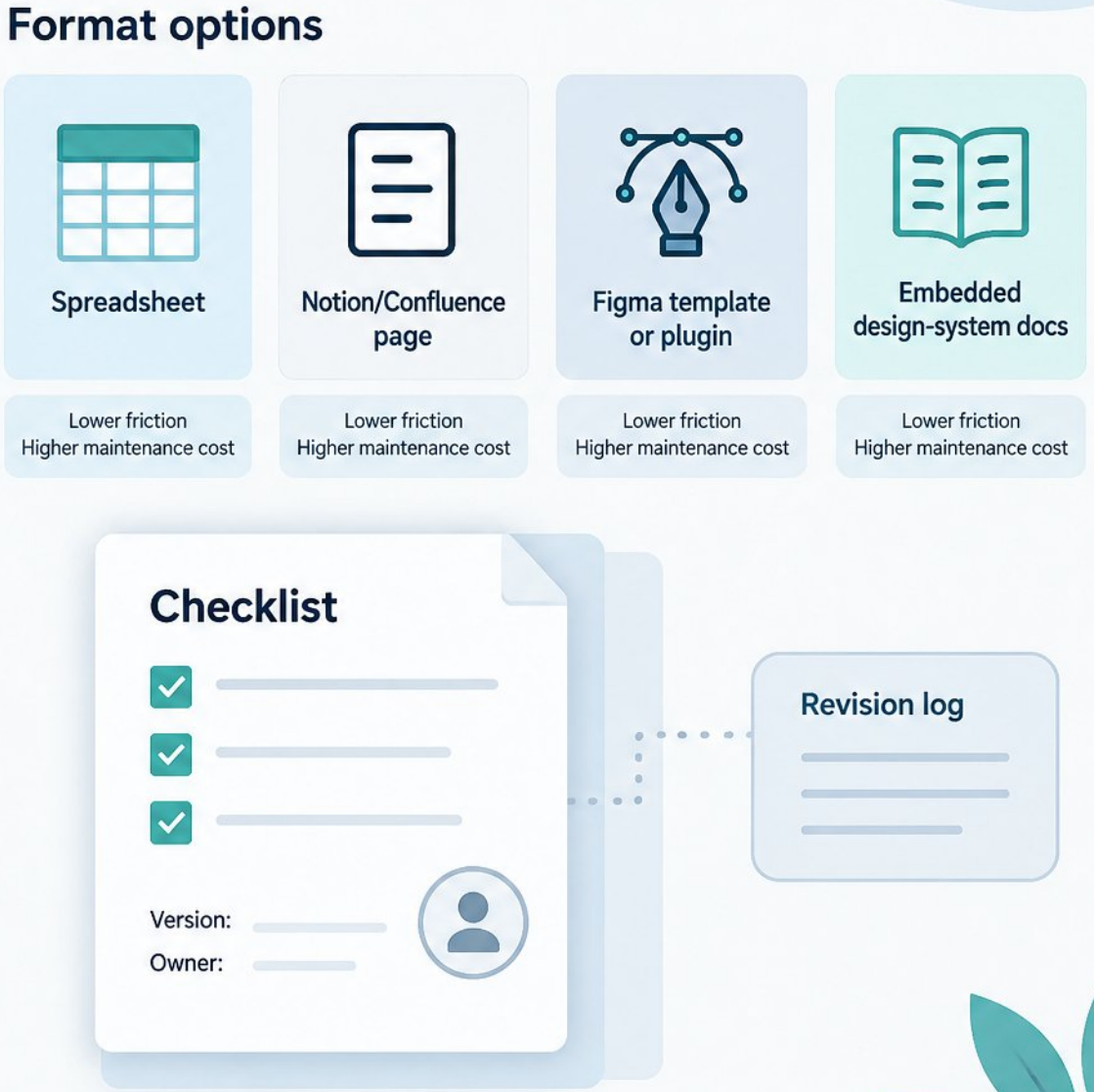
Accessibility Items Worth Stealing from Published Checklists

- Primer organizes WCAG 2.2 A/AA criteria by design concern, not by guideline number
- WCAG 2.2 added nine criteria; carry focus not obscured, target size 24×24, consistent help
- Translate criteria into Figma annotations: heading order, landmarks, alt text, focus order, labels
- Borrow structure, not volume: 55 A/AA criteria is a reference library
- Automation catches ~30–40% — make human and behavioral testing explicit checklist items



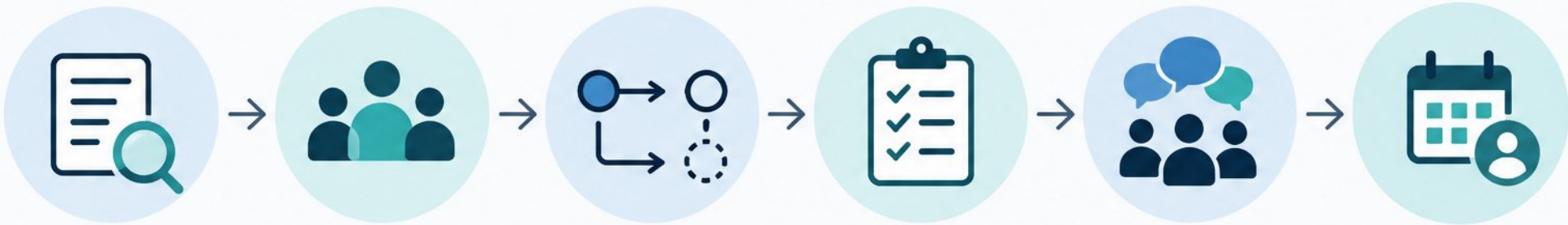
Formats, Tools, and Templates

- Format options: spreadsheet, Notion/Confluence page, Figma template or plugin, embedded design-system docs
- Real examples: Notion UX audit checklists and Figma annotation-and-marker review templates
- Integrate with issue trackers, handoff tools, accessibility scanners, and in-house review plugins
- Trade-offs: plugins lower friction but raise maintenance cost versus static documents
- Offer lightweight pre-handoff passes and heavyweight full audits
- Version the checklist; record which version scored each review



Running a Review with the Checklist

A clear process leads to better decisions and a better experience.



- **Pre-review:** scope the session, audience, artifacts, and success criteria
- **Roles:** designer, reviewer, facilitator, decision owner, plus one engineer
- Walk complete flows; give at least **30%** of review time to edge cases
- Link findings to checklist items and user impact; score severity 0–4
- **Use 3–5** independent evaluators, then reconcile disagreements
- **Log owner,** next step, and follow-up date; never reward comment volume

Adapting Checklists for Designers and Product Teams



Designers

- Designers: pre-handoff self-check on completeness, states, annotations, handoff readiness



Product teams

- Product teams: requirements-alignment and PRD coverage checkpoints tied to outcomes

Shared decision states

- approved
- approved with minor changes
- needs revision

- Split ownership: PM leads requirements + user flows; design lead leads system + handoff
- Record decisions as states: approved, approved with minor changes, needs revision
- **Shared vocabulary cuts handoff friction and rework**

Adapting Checklists for Reviewers, Educators, and Content Creators

- **Reviewers:** require evidence, calibrate severity, score independently before discussion
- **Educators:** convert items into rubrics with criteria, scoring levels, and observable look-fors
- **ISTE Teacher Ready precedent:** dimensions → indicators → observable look-fors
- **Rubrics state** criteria, scoring, and verdict format — plus what they cannot judge
- **Content creators:** structure, readability, tone, localization, alt text, captioning
- **One shared vocabulary** across all five roles prevents divergent standards



RUBRIC FRAMEWORK

Criteria	Scoring levels	Observable look-fors
What to look for	How to rate	Evidence to check

Measuring Effectiveness and Iterating

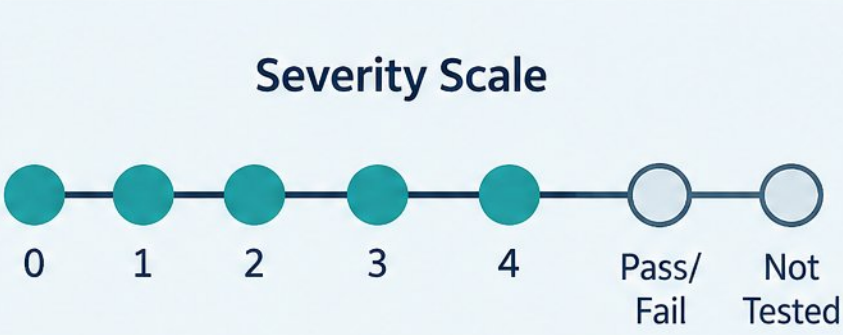
Leading indicators:
criteria-linked findings, recurring failures, post-release escapes, false positives

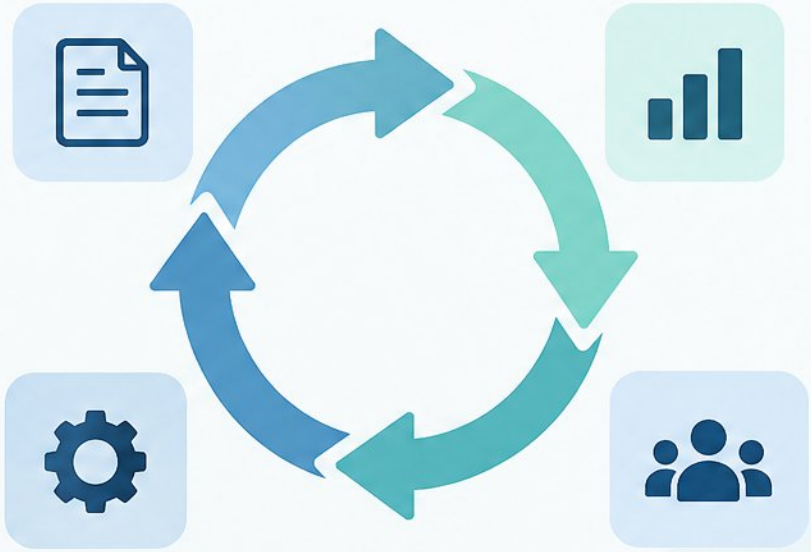
Efficiency indicators:
review cycle time, reviews within target window, team adoption

Agoda example:
83% cited quality gains;
90% saw no velocity loss

Calibrate severity with real artifacts; retire weak items, add with evidence

Assign a named owner and review cadence within design-system governance





Common Pitfalls and Practical Guardrails

Avoid common failure modes, and put practical guardrails in place.



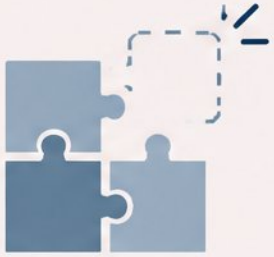
Checklist bloat drives shallow box-ticking; cap items and use tiered lists



Duplicating design-system rules instead of linking to them adds drift



Subjective taste as criterion, or severity drifting between reviewers, erodes trust



Late accessibility, localization, and content-state coverage costs rework



Tiered lists



Link to design-system rules



Explicit ownership and review cadence



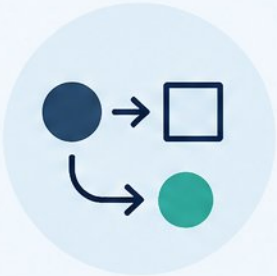
Version history and exceptions log



Guardrails: tiers, explicit ownership, review cadence, version history, exceptions log

Workshop: Draft Your First Checklist

Practice, collaborate, and build confidence with accessible, effective checklists.



Exercise 1

map one product flow to coverage categories



Exercise 2

write five well-formed items with rationale and evidence



Exercise 3

peer review a sample and calibrate severity together



Exercise 4

plan a 30-day adoption experiment with metrics and owner



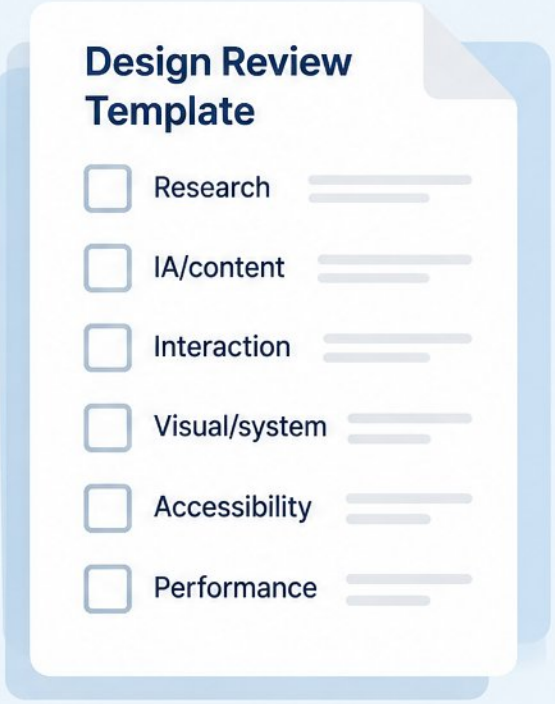
- [ ] Quick warmup: 5-second test, keyboard-only pass, contrast and 200% zoom
- [ ] Quick warmup: check empty and error states

Takeaways, Starter Template, and Next Steps

From review to real progress

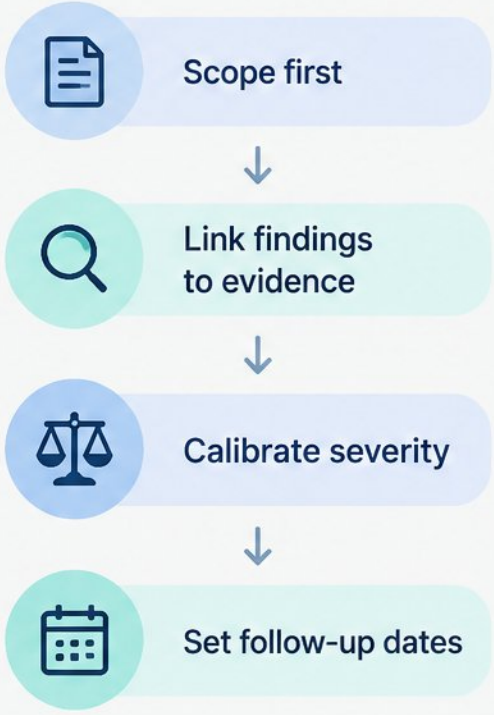
Starter Template

- Starter template spans research, IA/content, interaction, visual/system, accessibility, and performance



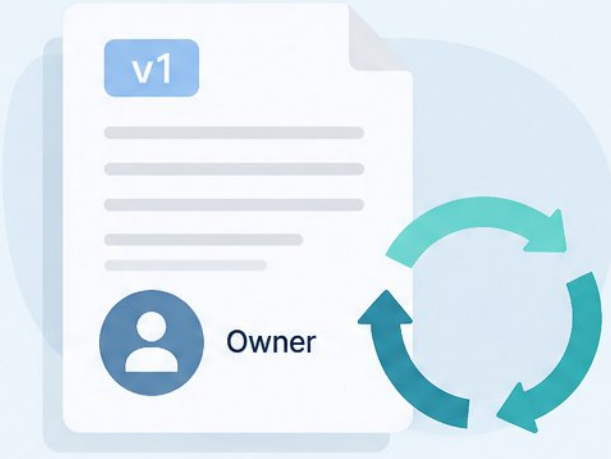
Review Ritual

- Review ritual: scope first, link findings to evidence, calibrate severity, set follow-up dates



Ownership and Next Steps

- Name an owner and a lightweight governance loop to keep iterating
- First 30 days: three reviews, record metrics, retire or rewrite two items
- Keep anchored to user goals, WCAG 2.2 AA, and Nielsen's heuristics



PersonWise.

PersonWise • AI digital-human course platform



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

personwise.ai/t/ed691664/public