H

Accessibility Audit Template

Comprehensive accessibility testing and reporting framework

Prepared by: AccessGuard

Version: 1.0

Date: October 18, 2025

Standards: WCAG 2.2 AA, Section 508

Disclaimer: This template supports accessibility audits but is not a substitute for professional legal advice.

Accessibility Audit Template

Table of Contents

[Back to TOC](#)

Use this template to conduct comprehensive audits of websites and applications. Combine automated scans with rigorous manual and assistive-technology testing.

1. Executive Summary

[Back to TOC](#)

Organization: _____
Website/Application: _____
Audit Date: _____
Auditor(s): _____
Audit Type: [ ] Full [ ] Partial [ ] Sampling

Overall Compliance Status

- WCAG 2.1 Level A: [ ] Pass [ ] Fail [ ] Partial
- WCAG 2.2 Level AA: [ ] Pass [ ] Fail [ ] Partial
- WCAG 2.2 Level AA: [ ] Pass [ ] Fail [ ] Partial

Summary Statistics

- Total Pages/Views Audited: _____

- Critical Issues: _____
- High Priority: _____
- Medium Priority: _____
- Low Priority: _____

2. Audit Scope

[Back to TOC](#)

Pages/Components Tested

- Homepage: [ ] Yes [ ] No
- Key User Flows (sign-in, checkout, etc.): [ ] Yes [ ] No
- Forms & Validation: [ ] Yes [ ] No
- Navigation/Header/Footer: [ ] Yes [ ] No
- Media (video/audio): [ ] Yes [ ] No
- Dynamic Widgets (modals, carousels): [ ] Yes [ ] No
- PDFs/Docs: [ ] Yes [ ] No

User Agents Tested

- Chrome [ ] Firefox [ ] Safari [ ] Edge [ ] iOS [ ] Android [ ]

Assistive Technologies

- NVDA (Windows) [ ]
- JAWS (Windows) [ ]
- VoiceOver (macOS/iOS) [ ]
- TalkBack (Android) [ ]

Testing Tools

- Automated: axe, Lighthouse, WAVE, Halytic verifier
- Manual: keyboard-only, focus order, error recovery, zoom 200–400%
- Color & Contrast: WCAG contrast analyzers

3. Methodology

[Back to TOC](#)

Automated Testing

- Tool(s): _____
- Date(s): _____
- Coverage: # of pages/components and selection method (sitemap/crawl, analytics-based)

Manual Testing

- Keyboard: Tab/Shift+Tab, Enter/Space, Escape, visible focus, traps
- Screen Reader: structure, landmarks, headings, alternative text, control labels
- Visual: color contrast, zoom/reflow, motion, flashing content
- Forms: labels, instructions, errors, autocomplete, programmatic relationships

User Testing (Optional)

- Participants with disabilities: [ ] Yes [ ] No — N = ___
- Scenarios: _____

4. Detailed Findings

Severity Definitions

- **Critical**: Blocks task completion or creates legal exposure across high-traffic flows
- **High**: Significant barrier; workaround exists but hard to discover
- **Medium**: Usability impairment; accessible alternative exists
- **Low**: Minor issue; does not block task completion

Findings Summary Table

ID	Page/Flow	Component	WCAG	Severity	Status	Notes
A-01				Critical	Open	
A-02				High	Open	
A-03				Medium	Open	

Example Criteria Breakouts

1.1.1 Non-text Content (Level A)

Status: [ ] Pass [ ] Fail [ ] Partial
Issues

1. **Missing or non-informative alt text**

- Page: _____
- Element: _____
- Impact: Screen reader users cannot understand imagery.
- **Recommendation**: Provide concise, contextual alt; mark decorative images as `alt=""`.

1.3.1 Info and Relationships (Level A)

- Incorrect heading hierarchy and unlabeled form groups.
- **Recommendation**: Use semantic HTML (H1–H6), `<fieldset>/<legend>`, and associated labels.

1.4.3 Contrast (Minimum) (Level AA)

- **Current Ratio**: ____ : 1 vs. **Required**: 4.5 : 1 (text); 3 : 1 (UI components).
- **Recommendation**: Adjust color tokens; provide design system contrast rules.

2.1.1 Keyboard (Level A)

- **Keyboard trap in modal**; missing focus return.
- **Recommendation**: Trap focus within modal; restore focus to invoking control on close.

2.4.3 Focus Order (Level A) / 2.4.7 Focus Visible (AA)

- Illogical focus progression; low-visibility outline removed.
- **Recommendation**: Logical DOM order and visible, consistent focus styles.

3.3.1 Error Identification (Level A) / 3.3.3 Error Suggestion (AA)

- Vague error messages; no programmatic association.
- **Recommendation**: Associate error text to fields via `aria-describedby`; provide specific correction advice.

4.1.2 Name, Role, Value (Level A)

- Custom widgets lack accessible names/roles.
- **Recommendation**: Use native elements or ARIA patterns; synchronize state with `aria-*` attributes.

(Continue criteria as needed for audit depth.)

5. Recommendations

[Back to TOC](#)

Priority 1: Critical (Immediate)

1. Fix keyboard traps in modal/dialog components.
2. Provide alt text for non-decorative images on top 20 pages.
3. Restore visible focus styles across interactive controls.

Priority 2: High (≤ 30 days)

1. Correct heading structure and landmark regions.
2. Resolve contrast failures in primary/secondary buttons and links.
3. Associate form inputs with labels and error messages.

Priority 3: Medium (≤ 90 days)

1. Refactor custom components to follow ARIA Authoring Practices.
2. Update color tokens in design system with contrast constraints.
3. Add automated checks to CI (axe, Halytic verifier).

Priority 4: Low (Next Quarter)

1. Provide transcripts and audio descriptions for legacy media.
2. Add skip links and "Back to top" anchors across templates.
3. Expand user testing with assistive-tech users.

6. Remediation Roadmap

[Back to TOC](#)

Phase	Timeline	Focus Areas	Owner
Phase 1	Weeks 1–2	Critical issues	
Phase 2	Weeks 3–4	High priority	
Phase 3	Month 2	Medium priority	
Phase 4	Month 3	Low priority	

7. Verification & Evidence

[Back to TOC](#)

- **Artifacts**: before/after HTML, diffs, screenshots, test logs
- **Assistive Tech Re-test**: NVDA + VoiceOver scenario validation
- **Regression**: Add Playwright/axe checks to CI pipeline

8. Sign-off

[Back to TOC](#)

Auditor: _____ Date: _____
Technical Lead: _____ Date: _____
Executive Sponsor: _____ Date: _____

Next Audit Date: _____