Website Accessibility

Implementation Guidelines

Department of Information Technology (DoIT)

www.doit.gov.np

Prepared by:

Internet Society Nepal Chapter

December 2025

Open Internet Nepal | Internet Society Nepal Chapter

Table of Contents

Executive Summary

This document provides comprehensive implementation guidelines for addressing accessibility issues identified in the Department of Information Technology (DoIT) website audit. The guidelines are based on Web Content Accessibility Guidelines (WCAG) 2.2 standards and prioritize improvements that will have the greatest impact on users with disabilities.

The implementation is organized into four priority levels, with clear technical specifications, code examples, and acceptance criteria for each improvement. This phased approach ensures that critical accessibility barriers are addressed first while building toward full WCAG 2.2 Level AA compliance.

Document Purpose

This implementation guide serves to:

1. Provide detailed technical specifications for each accessibility improvement
2. Offer code examples and best practices for implementation
3. Establish clear acceptance criteria for testing and validation
4. Define a prioritized roadmap for accessibility improvements
5. Ensure sustainable accessibility practices for future development

Implementation Priorities

The accessibility improvements have been categorized into four priority levels based on their impact on users with disabilities and compliance requirements. Each priority level builds upon the previous one, creating a sustainable path to full accessibility.

Priority 1: Critical Accessibility Barriers

Timeline: Implement within 2-4 weeks

These are fundamental issues that prevent users with disabilities from accessing basic website functionality. Addressing these items is essential for basic WCAG Level A compliance.

1.1 Skip Navigation Links

Issue: Keyboard and screen reader users must tab through all navigation items to reach main content.

WCAG Criteria: 2.4.1 Bypass Blocks (Level A)

Impact: Keyboard users waste significant time navigating through repetitive elements on every page.

Implementation:

Add a skip link as the first focusable element in the page header. The link should be visible when focused and should jump directly to the main content area.

Code Example (HTML):

```html
<a href="#main-content" class="skip-link">Skip to Main Content</a>

<!-- Main content area -->
<main id="main-content" tabindex="-1">
  <!-- Page content -->
</main>
```

Code Example (CSS):

```css
.skip-link {
  position: absolute;
  top: -40px;
  left: 0;
  background: #1F4788;
  color: white;
  padding: 8px 16px;
  text-decoration: none;
  z-index: 100;
}

.skip-link:focus {
  top: 0;
}
```

Acceptance Criteria:

- Skip link is the first element in tab order
- Link becomes visible when focused via keyboard
- Clicking the link moves focus to main content
- Works correctly in all major browsers and screen readers

1.2 Keyboard-Accessible Navigation Menu

Issue: Navigation menu and submenus are only accessible via mouse, preventing keyboard and screen reader users from accessing important site sections.

WCAG Criteria: 2.1.1 Keyboard (Level A), 4.1.2 Name, Role, Value (Level A)

Implementation Requirements:

- All menu items must be accessible via keyboard (Tab, Arrow keys, Enter, Escape)
- Proper ARIA attributes for menu structure and state
- Visual focus indicators on all interactive elements
- Screen reader announcements for menu state changes

Code Example (HTML Structure):

```html
<nav aria-label="Main Navigation">
  <ul role="menubar" aria-label="Main Menu">
    <li role="none">
```

```
    <a href="/home" role="menuitem">Home</a>
  </li>
  <li role="none">
    <button role="menuitem"
            aria-expanded="false"
            aria-haspopup="true">
      Services
    </button>
    <ul role="menu" hidden>
      <li role="none">
        <a href="/service-1" role="menuitem">Service 1</a>
      </li>
    </ul>
  </li>
  </ul>
</nav>
```

Acceptance Criteria:

- All menu items accessible via Tab key
- Submenus open with Enter/Space and close with Escape
- Arrow keys navigate within submenus
- Screen readers announce menu structure and expanded/collapsed state
- Visible focus indicators on all menu items

1.3 Form Labels and Descriptions

Issue: Form fields (search input, contact forms) lack proper labels, making them unusable for screen reader users.

WCAG Criteria: 3.3.2 Labels or Instructions (Level A), 4.1.2 Name, Role, Value (Level A)

Code Example (Search Form):

```
<form role="search" aria-label="Site Search">
  <label for="search-input" class="sr-only">Search website</label>
  <input type="search"
         id="search-input"
         placeholder="Search..."
         aria-label="Search website">
  <button type="submit" aria-label="Submit search">
    <span class="icon-search" aria-hidden="true"></span>
  </button>
</form>
```

Acceptance Criteria:

- Every form field has a visible or accessible label
- Labels are programmatically associated with their inputs
- Required fields are clearly marked and announced to screen readers
- Error messages and hints are accessible and announced

1.4 Alternative Text for Images and Icons

Issue: Logos, icons, service images, official photos, and gallery images lack descriptive alternative text.

WCAG Criteria: 1.1.1 Non-text Content (Level A)

Guidelines for Writing Alt Text:

- Be specific and concise (aim for 125 characters or less)
- Avoid phrases like "image of" or "picture of"
- Describe the purpose or function, not just appearance
- For decorative images, use empty alt text (alt="") or aria-hidden="true"
- For complex images, provide longer descriptions via aria-describedby

Code Examples:

Logo:

```
<img src="doit-logo.png"
     alt="Department of Information Technology logo">
```

Social Media Icons:

```
<a href="https://facebook.com/doit" aria-label="Visit DoIT on Facebook">
  <span class="icon-facebook" aria-hidden="true"></span>
</a>
```

Contact Icons:

```
<div class="contact-info">
  <span class="icon-phone" aria-hidden="true"></span>
  <span>Phone: </span>
  <a href="tel:+9771234567" aria-label="Call DoIT at 977-1-234567">
    977-1-234567
  </a>
</div>
```

Acceptance Criteria:

a. All meaningful images have descriptive alt text
b. Decorative images use empty alt text or aria-hidden
c. Icons inside links have proper aria-labels on the link
d. Screen readers announce image purpose correctly

Priority 2: Content Accessibility

Timeline: Implement within 4-6 weeks

These improvements ensure that website content is accessible and understandable to all users, particularly those using assistive technologies.

2.1 Accessible Document Formats

Issue: Documents, notices, publications, and resources are only available as PDFs, which are often inaccessible to screen reader users.

WCAG Criteria: 1.1.1 Non-text Content (Level A)

Implementation Options:

Option 1: HTML Versions (Recommended)

Provide HTML versions of all documents directly on the website. This is the most accessible format and allows users to navigate with screen readers, adjust text size, and use browser assistive features.

Option 2: Accessible PDFs

If PDFs must be used, ensure they are properly tagged and accessible:

a. Add document title and language
b. Use proper heading structure (H1, H2, H3)
c. Add alt text to all images and charts
d. Ensure reading order is correct
e. Use actual text (not scanned images)
f. Test with PDF accessibility checker

Acceptance Criteria:

a. All new documents published in HTML format
b. Existing important documents converted to HTML within 3 months
c. If PDFs are used, they pass accessibility checkers
d. Screen readers can navigate document structure

2.2 Logical Heading Structure

Issue: Heading hierarchy is inconsistent across pages, making navigation difficult for screen reader users.

WCAG Criteria: 1.3.1 Info and Relationships (Level A), 2.4.6 Headings and Labels (Level AA)

Heading Structure Rules:

a. Use only one H1 per page (typically the page title)
b. Don't skip heading levels (e.g., H1 to H3)
c. Headings should reflect content hierarchy
d. Don't use headings just for visual styling

Code Example (Homepage Structure):

```
<h1>Department of Information Technology</h1>

<section>
  <h2>Our Services</h2>
  <div class="service">
    <h3>Digital Governance</h3>
```

```
    <p>Description...</p>
  </div>
</section>

<section>
  <h2>Latest News</h2>
  <article>
    <h3>Digital Nepal Framework Launched</h3>
    <p>News content...</p>
  </article>
</section>
```

Acceptance Criteria:

a. Each page has exactly one H1
b. Heading levels follow logical order without skipping
c. Screen reader users can navigate page structure via headings

2.3 Clear and Descriptive Links

Issue: Links lack context (generic text like "Read More", "Click Here"), and users aren't warned when links open in new windows.

WCAG Criteria: 2.4.4 Link Purpose (In Context) (Level A), 3.2.2 On Input (Level A)

Link Best Practices:

• Link text should describe the destination or purpose
• Avoid generic text like "Click Here," "Read More," "Learn More"
• Combine multiple redundant links into one
• Warn users when links open in new windows/tabs

Code Example (News Links - Correct):

```
<article class="news-item">
  <a href="/news/1" class="news-link">
    <img src="news1.jpg" alt="">
    <h3>Digital Literacy Program Launches in 50 Schools</h3>
    <p>The Department of Information Technology announced...</p>
  </a>
</article>
```

Code Example (External Links):

```
<a href="https://example.com"
   target="_blank"
   rel="noopener noreferrer">
  External Resource
  <span class="sr-only">(opens in new tab)</span>
  <span aria-hidden="true" class="icon-external"></span>
</a>
```

Acceptance Criteria:

• All links have descriptive, meaningful text
• No generic link text like "Click Here" or "Read More"
• Redundant links combined into single links
• Links that open in new windows include warning text

Priority 3: Enhanced Usability

Timeline: Implement within 6-8 weeks

These improvements enhance the user experience for people with visual impairments and cognitive disabilities.

3.1 Sufficient Color Contrast

Issue: Poor contrast between text and background colors makes content difficult to read.

WCAG Criteria: 1.4.3 Contrast (Minimum) (Level AA)

Contrast Requirements:

a. Normal text: 4.5:1 contrast ratio minimum
b. Large text (18pt+ or 14pt+ bold): 3:1 contrast ratio minimum
c. Use tools like WebAIM Contrast Checker to verify colors

Recommended Color Palette:

Use Case	Color	Contrast Ratio
Primary text	#212121 on white	16.1:1
Secondary text	#595959 on white	7.0:1
Links	#1F4788 on white	8.6:1

Acceptance Criteria:

a. All text meets minimum 4.5:1 contrast ratio
b. Large text meets minimum 3:1 contrast ratio
c. Verify with automated contrast checking tools

3.2 Visible Focus Indicators

Issue: Inconsistent or missing visual indicators make it difficult for keyboard users to track their position on the page.

WCAG Criteria: 2.4.7 Focus Visible (Level AA)

Code Example (CSS):

```css
/* Custom focus styles for all interactive elements */
a:focus,
button:focus,
input:focus,
select:focus,
textarea:focus,
[tabindex]:focus {
  outline: 3px solid #1F4788;
  outline-offset: 2px;
}
```

Acceptance Criteria:

a. All interactive elements have visible focus indicator
b. Focus indicator has sufficient contrast (3:1 minimum)
c. Focus order follows logical reading order

3.3 Accessible Status Messages and Notifications

Issue: Form submission confirmations and error messages are not announced to screen reader users.

WCAG Criteria: 4.1.3 Status Messages (Level AA)

Code Example:

```
<!-- ARIA live region for notifications -->
<div id="notification-region"
     role="status"
     aria-live="polite"
     aria-atomic="true"
     class="sr-only">
</div>
```

Acceptance Criteria:

a. Success messages announced to screen readers
b. Error messages associated with form fields
c. Status changes communicated without requiring focus change

Priority 4: Advanced Accessibility Features

Timeline: Implement within 8-12 weeks

These enhancements provide additional support for users with diverse abilities and ensure full WCAG 2.2 Level AA compliance.

4.1 Multimedia Accessibility

Recommendation: When adding videos to the website, include Nepali sign language interpretation and captions for deaf and hard-of-hearing users.

Requirements:

a. Provide captions/subtitles in Nepali and English
b. Include Nepali sign language interpretation
c. Provide audio descriptions for visual content
d. Include transcript of video content in HTML format

4.2 Accessible Touch Target Sizes

Recommendation: Create larger touch areas for buttons and interactive elements to accommodate users with motor disabilities.

Minimum Size Requirements:

a. Interactive elements: minimum 44x44 pixels

b. Adequate spacing between clickable elements
c. Touch targets don't overlap

4.3 Accessible Chatbot Interface

Issue: The chatbot widget in the footer is not accessible to users with assistive technologies.

Requirements:

a. Keyboard accessible (open, close, navigate messages)
b. Screen reader announcements for new messages
c. Proper ARIA labels and roles
d. Clear focus management

Testing and Validation

Each accessibility improvement must be thoroughly tested before deployment. Use a combination of automated tools and manual testing with actual users who have disabilities.

Automated Testing Tools

Recommended Tools:

a. WAVE Web Accessibility Evaluation Tool (browser extension)
b. axe DevTools (browser extension)
c. Lighthouse (built into Chrome DevTools)
d. WebAIM Contrast Checker (https://webaim.org/resources/contrastchecker/)
e. Accessibility Checker (https://www.accessibilitychecker.org/)

Manual Testing Procedures

Keyboard Testing

Navigate the entire website using only the keyboard:

a. Tab through all interactive elements
b. Verify all functionality is accessible without a mouse
c. Check that focus indicators are clearly visible
d. Ensure logical tab order throughout pages

Screen Reader Testing

Test with common screen readers:

a. NVDA (Windows - free and open source)
b. JAWS (Windows - commercial)
c. VoiceOver (macOS/iOS - built-in)

Verify:

a. All images have meaningful alt text
b. Headings structure is announced correctly
c. Form labels are associated with inputs
d. Link purposes are clear from the link text
e. Navigation menus work properly

User Testing with People with Disabilities

Conduct usability testing sessions with actual users who have disabilities. This is the most valuable form of testing and will reveal issues that automated tools cannot detect.

Recommended Testing Groups:

a. Screen reader users (blind or low vision)
b. Keyboard-only users (motor disabilities)
c. Users with cognitive disabilities
d. Deaf or hard-of-hearing users

Implementation Roadmap

The following timeline provides a structured approach to implementing all accessibility improvements over a 12-week period.

Week	Priority Level	Tasks
1-2	Priority 1	Implement skip links, keyboard navigation, form labels
3-4	Priority 1	Add alt text to all images, icons, and graphics
5-6	Priority 2	Begin converting PDFs to HTML, fix heading structure
7-8	Priority 2	Improve link descriptions, remove redundant links
9-10	Priority 3	Fix color contrast issues, add focus indicators, implement status messages
11-12	Priority 4	Implement multimedia accessibility features, improve touch targets, make chatbot accessible

Note: Testing should be conducted continuously throughout the implementation process, not just at the end. Each completed feature should be validated before moving to the next priority level.

Maintenance and Ongoing Compliance

Accessibility is not a one-time project but an ongoing commitment. Establishing processes for maintaining accessibility ensures that the website remains accessible as content is added and updated.

Content Guidelines for Editors

All content editors and contributors should follow these accessibility guidelines when adding or updating content:

When Adding Images:

- Always provide descriptive alt text
- Keep alt text concise but informative
- Use empty alt text for decorative images

When Creating Content:

- Use proper heading hierarchy
- Write descriptive link text
- Ensure sufficient color contrast
- Provide text alternatives for multimedia

When Adding Documents:

- Prefer HTML over PDF
- If using PDF, ensure it's tagged and accessible
- Provide alternative formats when possible

Regular Accessibility Audits

Conduct accessibility audits on a regular schedule:

- Monthly: Automated testing with WAVE, axe DevTools
- Quarterly: Manual keyboard and screen reader testing
- Annually: Comprehensive audit by accessibility experts
- Ongoing: User feedback collection from people with disabilities

Training and Awareness

Ensure all team members understand accessibility:

- Provide accessibility training for all content editors
- Include accessibility in developer onboarding
- Share success stories and impact of accessibility improvements
- Create internal accessibility champion program

Resources and References

WCAG Guidelines
Web Content Accessibility Guidelines (WCAG) 2.2
https://www.w3.org/WAI/WCAG22/quickref/

Testing Tools
WAVE: https://wave.webaim.org/

axe DevTools: https://www.deque.com/axe/devtools/

WebAIM Contrast Checker: https://webaim.org/resources/contrastchecker/

Accessibility Checker: https://www.accessibilitychecker.org/

Screen Readers
NVDA (Windows): https://www.nvaccess.org/

JAWS (Windows): https://www.freedomscientific.com/products/software/jaws/

VoiceOver (macOS/iOS): Built into Apple devices

Additional Resources
- WebAIM: https://webaim.org/
- A11y Project: https://www.a11yproject.com/
- MDN Accessibility: https://developer.mozilla.org/en-US/docs/Web/Accessibility
- Deque University: https://dequeuniversity.com/

Appendix: Quick Reference Checklist

Use this checklist to verify accessibility improvements:

Navigation

- ☐ Skip to main content link present and functional
- ☐ All navigation accessible via keyboard
- ☐ Focus indicators visible on all interactive elements
- ☐ Menu states announced to screen readers

Forms

- ☐ All form fields have labels
- ☐ Required fields clearly marked
- ☐ Error messages accessible and descriptive
- ☐ Success confirmations announced

Images and Media

- ☐ All images have alt text
- ☐ Decorative images marked as decorative
- ☐ Icons have accessible labels
- ☐ Videos have captions and transcripts

Content

- ☐ Headings follow logical hierarchy
- ☐ Links have descriptive text
- ☐ Text has sufficient contrast
- ☐ Documents available in HTML format

Conclusion

Implementing these accessibility improvements will make the Department of Information Technology website usable by all citizens, including those with disabilities. This phased approach ensures that critical barriers are addressed first while building toward full compliance with international accessibility standards.

By following these implementation guidelines, DoIT will:

- Ensure equal access to government services for all citizens
- Comply with WCAG 2.2 Level AA standards
- Demonstrate leadership in digital accessibility in Nepal
- Create a sustainable framework for ongoing accessibility
- Improve user experience for all website visitors

The commitment to accessibility is a commitment to digital equality. With the dedicated efforts of the development team and ongoing support from accessibility experts and users with disabilities, the DoIT website can become a model of accessible government service delivery.