

# Designing Websites for All Users: Resource List

Janet Sylvia, Web Accessibility Group (WAG) Leader  
Rock Eagle 2014

## Introduction to Web Accessibility

- [USG Accessibility Policy](#)
- [WebAIM Section 508 Checklist](#)
- [WebAIM WCAG 2.0 Checklist](#)

## 10 Tips for Creating Accessible Websites

### 5 Tips for Content Managers

1. Page Titles, Headings, Semantic Structure
  - a. <title> should match Heading 1 <h1>
  - b. Headings properly nested
  - c. <ol> for Ordered lists and <ul> for Unordered Lists
  - d. <strong> and <em>
2. Descriptive Hyperlinks
  - a. Link text makes sense when read out of context
  - b. Describes destination (website or document title)
  - c. Unique for each unique hyperlink
3. Alternative Text for all Non-text Content
  - a. Writing Effective Alt Text (Recorded Training), please contact Janet Sylvia [jsylvia@uga.edu](mailto:jsylvia@uga.edu) for access to this recording
  - b. Every image requires Alt text
  - c. Long Description when Alt alone is not enough
  - d. Image Carousels:
    - i. [Accessibility Expert Warns – Stop Using Image Carousels](#)
    - ii. [AccessIQ: Anatomy of an Accessible Carousel](#)
    - iii. [W3C Tutorials: Sliders/Carousel Concepts](#)
4. Accessible Documents
  - a. [WAG](#), under Resources, Training, several resources for creating accessible documents
  - b. [USG Accessibility Tutorial, Training, Intermediate](#) (HHS Accessible Documents Checklists)
5. Accessible Multimedia
  - a. Audio-only (Text Transcript)
  - b. Video-only (Video Description)
  - c. Audio+Video (Captions, Text Transcript, Video Description)
  - d. Accessible Media Player (see below)

## 5 Tips for Designers and Developers

1. Don't auto-play video
  - a. Provide controls to stop/pause
  - b. [YouTube HTML5 Video Player](#)(to request accessible HTML5 player)
2. Ensure JavaScript functionality is device independent
  - a. [WebAIM: Accessible JavaScript](#)
3. Ensure keyboard accessibility
  - a. Focus Indicators visible via Tab
    - i. Visit [WebAIM](#) site as a good example of how keyboard accessibility should work: use Tab (move forward), Shift+Tab (move backward) and Enter to ensure access to all navigation elements, widgets, content, form controls, etc.
  - b. Navigation Order - logical and intuitive
  - c. Interactive elements accessible via keyboard
  - d. Scripted elements and widgets accessible via keyboard
  - e. Lengthy navigation needs Skip to main content, Headings, [ARIA landmarks](#)
  - f. For assistance remediating keyboard access problems, please review [WebAIM: Keyboard Accessibility](#)
4. Provide sufficient contrast
  - a. [Colour Contrast Analyser](#)
5. Provide Accessibility Statement and contact information
  - a. The [USG Institution] complies with Section 508 and WCAG 2.0 for website accessibility. Please contact us if you cannot access information on this website.
  - b. Email Address
  - c. Phone Number
  - d. Respond to any requests within 24 hours or 1 business day

## Accessibility Checkers

- Toolbars:
  - [Web Accessibility Toolbar \(WAT\)](#)
  - [Firefox Web Developers Toolbar](#) + Accessibility Extension
  - Use with [Firefox Accessibility Evaluation Toolbar](#)
  - [WAVE Toolbar](#) (works on password protected pages, dynamically generated content, etc.
  - Step-by-step instructions for using each toolbar:
    - [W3C Easy Checks – A First Review of Web Accessibility](#)
- Automated Checkers (Free):
  - [WAVE](#) single pages
  - [Functional Accessibility Evaluator \(FAE\)](#), multiple levels, web crawling