Accessible Content Management with Adobe Experience Manager

Kiran Kaja
What is Accessibility?

- Accessibility involves two key issues:
  - How users with disabilities access electronic information
  - How content designers, developers, and authors produce content that functions with assistive devices used by individuals with disabilities.
- Accessibility is not a feature, it’s about procedures, processes, and techniques
The Importance of Best Practices

- Accessibility is NOT a Feature, it’s a Result
  - There is NO Accessibility Button – Accessible Content Creation is a Process NOT a Feature
- Achieving Accessibility Requires Human Testing in addition to Automated Checking
- Checking Can Only Detect for the Presence or Lack of Required Items
- Cannot Check if an Item is Correct or Appropriate
Challenges (Where you probably are right now)

- You have a zillion web pages and/or documents
  - That aren’t actually documents, but a bunch of fragments thrown together
  - That 10 or 100 or 1000 or 10000 people can edit
  - Of which you’re one of 3 who know what they’re doing
- You have a CMS
  - Which came with templates you threw away
  - Probably developed by a consulting firm
  - And you don’t know what they did
- You have an accessibility problem
  - And somebody probably told your CIO their tool can solve it
Remediation Steps (What to do)

• Look at your top pages on your top sites
• Fix the most popular, most broken content
• Solve template problems first
• Minimize errors entering the system
• Train your users…just a little
• Establish standards for
  • Design
  • Scripting
• Give yourself some room
• Always focus on people
Web Content Accessibility Guidelines (WCAG) 2.0

- A set of technology agnostic Accessibility guidelines developed by W3C
  - http://www.w3.org/TR/WCAG/
- Supported by non-normative documents
  - Understanding WCAG 2.0
    - http://www.w3.org/TR/UNDERSTANDING-WCAG20/
  - Techniques for WCAG 2.0
    - http://www.w3.org/TR/WCAG20-TECHS/
- 3 levels of conformance: Level A, Level AA & Level AAA
  - Level AA is realistic & widely used/accepted
  - WCAG2 being used as basis of legislation
  - Latest Section 508 updated draft in the US, Canada, Australia, EU, etc
• Producing Accessible Sites and Applications with AEM

• http://dev.day.com/docs/en/cq/current/administering/supporting-accessibility.html

• Guide explains how to meet WCAG 2.0 success Criteria using AEM.
Prerequisites: Administrators

- Rich Text Editor
  - Install & configure the paraformat plugin to enable formatting options.
  - H1 through H6, lists, paragraphs, etc
  - Also add any block level semantic elements that are not available by default.
  - Enables administrators to specify additional HTML tags/attributes that can be used by content authors
Prerequisites: Administrators

- Decide on the formats styles that content authors can use: paragraph, h1, h2, etc
- Then specify the paragraph formats available in drop-down list of RTE
- Formats can be added as nodes under the RTE Plugins/paraformat node
Prerequisites: Administrators

• Install the “Enable All RTE Features” Package
• Provides a default set of formats and styles that can be further configured
• Also adds a source edit mode for modifying resulting HTML
• Download the package from the support site: http://dev.day.com/docs/en/cq/current/administering/package_manager.html#Package%20Share
Providing Text Alternatives

- Provide meaningful alt text for static graphics & images used as interactive components
- Image component dialog box > Advanced image properties tab > alt text
- If the image is decorative, use a space character in the alt text field to inform screen readers to ignore the image
- For complex images such as pie charts:
  - Provide a short explanation in alt text
  - Provide more detailed information in description field
Providing Text Alternatives

- Advanced Image Properties dialog

![Advanced Image Properties dialog](image-url)
Appropriate Structural Elements

- Use appropriate structural element for your page content in the Rich Text Editor
- `<p>` for paragraphs, `<ul>` or `<ol>` for lists, etc
- `<strong>` or `<em>` for bold and emphasised text
- Use format menu in RTE to pick correct structural element
Using Headings

- Create structure to your web pages by adding section headings
- If RTE Features Package is installed, H1, H2 & H3 are already available (Refer to Slide 20)
- Additional heading levels (H4 through H6 can be configured by administrators (Refer to Slide 20)
- Correctly nested headings help screen reader users navigate content easily
- Do not use headings to provide simple emphasis, use <strong> or <em>
Using Lists

• All 3 HTML list types are supported:
• Ordered, unordered & definition lists
• Select the list type from the format menu
• Using lists correctly provides additional navigation capabilities for screen reader users
Using Tables

- Tables of data must be identified using HTML table elements:
  - One `<table>` element
  - A `<tr>` element for each row of the table
  - A `<th>` element for each row and column heading
  - A `<td>` element for every data cell
  - A `<caption>` element to display a visible caption for the table
  - A `<summary>` element to provide a synopsis of the table for non-sighted users
  - `<summary>` is not visually displayed
  - The scope attribute of `<th>` can be used to indicate that the cell is a header for a particular row or column
  - For complex tables, header and id attributes need to be used for explicit associations
Using Tables (Continued)

- Insert table in the Rich Text Editor
- Select the type of headers:
  - Top for column headers, left for rows or both
- Create or edit header cells by opening context menu > cell properties dialog
Page Titles

- Provide a meaningful page title for all HTML pages
- Specify the title when creating a new HTML page
- Edit the page title in the page properties dialog
Labels for Form Fields

- All form fields need to have meaningful labels
- To edit the default label “title” for a form component:
  - Open field properties for that component
  - Edit the label (Title) in the Title & Text tab
- Label (Title) can be hidden but only do this if absolutely necessary
- Most screen readers announce hidden labels
- For ImageButton components, modifying title modifies the alt text
Link Purpose and Context

• Bad example of link text:
  • click here for details of our evening classes for autumn 2010.

• Good example:
  • Evening classes for autumn 2010 – details.

• Screen readers can display list of links in a page for users to navigate
• Title attribute may be used for providing extra instructions
• Use of title attribute is not recommended because:
  • Text in title attribute is only available to mouse users
  • Assistive Technology support is inconsistent – title attribute recognition may be turned off by default
Resources

- Adobe Accessibility Resource Center
  adobe.com/accessibility

- Adobe Accessibility Blog
  blogs.adobe.com/accessibility

- Producing Accessible Sites and Applications using AEM:
Q&A