

ADA Compliance Guide

Sitecore Discover

ADA Compliance Guide Document V 3.0.2 20200714

Table of Contents

Chapter 1	ADA Overview	3
Chapter 2	Accessibility Standards	4
2.1	Interoperability	5
2.2	Forward and Backward Browser Compatibility	6
2.3	Support for Pervasive Computing Technologies	7
2.4	Accessibility	8
Chapter 3	Text Equivalents for Images and other Non-Text Objects Best Practices	9
3.1	Non-Text Elements	10
3.2	HTML Elements for Text Equivalents	11
Chapter 4	Scripting Best Practices	12
4.1	Major Issues	13
Chapter 5	Styling Best Practices Overview.....	14
Chapter 6	Sitecore Discover's ADA compliance support.....	15

Sitecore® is a registered trademark. All other brand and product names are the property of their respective holders. The contents of this document are the property of Sitecore. Copyright © 2001-2022 Sitecore. All rights reserved.

Chapter 1 ADA Overview

Accessibility checklist published by the U.S. Department of Health and Human Services:

- Every image, video file, audio file, plug-in, etc. has an `alt` tag
- Complex graphics are accompanied by detailed text descriptions
- The alt descriptions describe the purpose of the objects
- If an image is also used as a link, make sure the `alt` tag describes the graphic and the link destination
- Decorative graphics with no other function have empty alt descriptions (`alt= ""`)
- Add captions to videos
- Add audio descriptions
- Create text transcript
- Create a link to the video rather than embedding it into web pages
- Add a link to the media player download
- Add an additional link to the text transcript
- The page should provide alternative links to the Image Map
- The `<area>` tags must contain an alt attribute
- Data tables have the column and row headers appropriately identified (using the `<th>` tag)
- Tables used strictly for layout purposes do NOT have header rows or columns
- Table cells are associated with the appropriate headers (e.g. with the `id`, `headers`, `scope` and/or axis HTML attributes)
- Make sure the page does not contain repeatedly flashing images
- Check to make sure the page does not contain a strobe effect
- A link is provided to a disability-accessible page where the plug-in can be downloaded
- All Java applets, scripts and plug-ins (including Acrobat PDF files and PowerPoint files, etc.) and the content within them are accessible to assistive technologies, or else an alternative means of accessing equivalent content is provided
- When form controls are text input fields, use the `<label>` element
- When text is not available use the title attribute
- Include any special instructions within field labels
- Make sure that form fields are in a logical tab order
- Include a **Skip Navigation** button to help those using text readers

Chapter 2 Accessibility Standards

2.1 Interoperability

[W3C Web Standards](#) support interoperability and give all users more options in accessing and using web content. People with disabilities use a more diverse set of technologies to access the web than the general population.

2.2 Forward and Backward Browser Compatibility

Use of web standards supports forward looking web design. For example, as computer monitors increase in pixel density, many current web pages will shrink in size and eventually become unusable for all users. The use of web standards allows web pages to adapt to these and other changes in display technology.

2.3 Support for Pervasive Computing Technologies

The use of web standards allows cell phones, PDAs, and new voice browser technologies to access and use web resources without the need to develop special versions or use problematic conversion tools.

2.4 Accessibility

[W3C Web Content Accessibility Guidelines 2.2](#) and [Section 508](#) provide guidance in the HTML markup issues related to accessibility.

Chapter 3 Text Equivalents for Images and other Non-Text Objects Best Practices

While ensuring accessibility does not mean that the web resource should avoid using graphical elements, it is important that all non-text elements are accompanied by text equivalents. Non-text elements are not accessible to users relying on speech readers and braille displays. Some users may use text-based browsers that do not support non-text elements, and some may have turned off support for non-text elements. Thus, text equivalents should be rendered in such a way as to effectively substitute non-text elements for variety of browsers and assistive technologies employed by people with different disabilities. Text equivalents can also help all users find pages more quickly, since search robots can use the text when indexing the pages.

3.1 Non-Text Elements

Non-text elements that should be provided with a text equivalent include:

- Images
- graphical representations of text (including symbols)
- image map regions
- animations (e.g., animated GIFs)
- applets and programmatic objects
- ASCII art
- Frames
- Scripts
- images used as list bullets
- Spacers
- graphical buttons
- sounds (played with or without user interaction)
- stand-alone audio files
- audio tracks of video
- video

3.2 HTML Elements for Text Equivalents

alt attribute for img and area elements

`alt` attribute provides a short text equivalent to orient the user to the purpose and/or content of the image.

longdesc attribute for img and frame

`longdesc` attribute is poorly supported by graphical browsers and assistive technologies, making it difficult for web developers to rely on this method for providing longer descriptions of images. In general, longer descriptions should be integrated into the web page, ensuring that the information is available to everyone, including people with disabilities. The text, along with the image, provides people with the opportunity to use more than one way to understand the content.

object element

Ensure a text equivalent is provided through the content of the object.

Chapter 4 Scripting Best Practices

The creation of the web was driven by interoperability and device independence, so that anyone could access the web regardless of the operating systems or devices he/she is using. The web is becoming more interactive as application developers are using various tools to implement their desired interface and interactive functionalities. Problems arise with the automation mechanism used in static and dynamic pages as this makes interaction with the web extremely difficult and sometimes impossible. For example, the user is required to hover the mouse on a link or dynamic menu to be able to open it. Or, as soon as a selection is made in a combo box, the page refreshes automatically and new form controls appear or disappear, or the user is directed to a completely different page. All of these automatic changes make interaction with the page very difficult and sometimes impossible for disabled users.

4.1 Major Issues

- Keyboard support and focus management
- Orientation to keyboard shortcuts for web controls
- Information to states and properties of web controls
- Orientation to dynamic content

Chapter 5 Styling Best Practices Overview

The best practices recommend that web developers use CSS technologies as the primary way to style content rather than using inline markup elements to style text, inline images for spacing or decoration, and tables for layout. By using structured markup and CSS to style the structure, web developers will create web sites that are easier to maintain and create. The use of inline styling markup, inline images and tables for layout requires extensive editing to make changes and verify consistency with similar content. Using structured markup with CSS makes it much easier to have consistent and complete styling as new pages are built and existing pages modified, as the author does not need to view or edit the inline markup. This also makes pages much smaller and therefore faster to load than pages weighed down with extensive inline styling markup.

Chapter 6 Sitecore Discover's ADA compliance

support

U.S. Department of Health and Human Services has extensive ADA compliance guide. It is the customer's responsibility to determine what parts of the ADA compliance guide they wish to honor. Customers are expected to publish an ADA Compliance policy for their website. This policy must cover the full website. Sitecore Discover is committed to complying with the the published guide for the areas of the page that is powered by Discover.

The customer is expected to share the compliance guide with Sitecore Discover, and walk through key ADA compliance policies they adhere to.

The key support Discover provides automatically is for describing the image of the product. According to ADA, the alt text for images should describe the image of the product or content. We currently automatically generate alt text as "image of <PRODUCT NAME>" to meet this requirement. If you have more descriptive text of product images, it can be provided in the product feed, and and we can use it to replace our default alt text.

Similarly, we also provide the ability for customers to enter alt text for images shown for content personalization like banner and other content images. If alt text is not specified, Discover automatically generates the alt text as "image of <Banner Name>" or "image of <Content Widget Name>"

All other information displayed by Discover is textual and can be read by the ADA readers.