Project MUSE Accessibility Guide

April 2017

PROJECT MUSE

Made possible by a generous grant
from the
Andrew W. Mellon Foundation.

muse.jhu.edu

Table of Contents

Preface: Accessibility 101 for Publishers

Project MUSE is committed to making our publishers' content accessible to all researchers, regardless of their physical ability. To this end, we have worked hard to meet or exceed commonly accepted standards for digital accessibility.

Our Voluntary Product Accessibility Template (VPAT) documents our compliance with the US Federal standards found in Section 508 of the Rehabilitation Act. Additionally, this Web Content Accessibility Guidelines 2.0 (WCAG 2) checklist demonstrates that we are compliant with all of Level A and much of Level AA of these strict international accessibility standards.

In adhering to these standards, Project MUSE's web pages are designed to:

- Work with all assistive devices, including screen readers and braille readers, that are available at the creation of the document.
- Allow assistive devices to skip over repeated text (such as menus) to ease usage.
- Be navigated with only a keyboard and do not have any features that require a mouse to operate.
- Offer alternative text and captions (which are only visible to assistive devices) for all non-text elements, including icons and images.
- Feature text that can be enlarged by 200% or more.

In order to achieve our goal of being as accessible as possible to all researchers, we rely on our publishing partners' support and cooperation. In particular, we depend on publishers to submit content that meets certain

technical specifications that are designed to maximize accessibility. For example, we require that PDFs use fully embedded fonts so that all text can be read by assistive devices.

Images and videos pose special challenges for users who rely on assistive devices. Project MUSE has features to minimize these difficulties. For example, our video player will be updated to fully support closed captioning in May 2017.

However, our ability to deliver accessible content to researchers rests in large part on the content that we receive from publishers. We ask that you keep a few simple guidelines in mind when preparing your content for delivery to MUSE:

- High-resolution images work best when enlarged by users with limited visibility.
- Alternative captions (AKA “rich captions”) give a person using an assistive device an experience closer to that of a sighted user by describing images in more detail than do print captions.
- Captions and/or transcripts for video and audio supplements greatly enhance access for visually- and hearing-impaired researchers. Video and audio supplements without captions and/or transcripts are often almost entirely inaccessible.

The following guide will outline how you can help Project MUSE make your content accessible to the widest audience possible.

Part 1:
Introduction

United States Accessibility Laws: Historical Highlights

1920: **Smith-Fess Act** provides for joint federal-state vocational programs for people with physical disabilities.

1943: **Barden-LaFollette Act** augments the 1920 law and adds limited services for people with mental disabilities.

1973: Congress passes the **Rehabilitation Act of 1973**, Section 504 of which prohibits organizations that receive federal funds from discriminating against "otherwise qualified handicapped" individuals.

1975: **Individuals with Disabilities Education Act (IDEA)** guarantees disabled children the right to a free, appropriate education in the least restrictive environment possible.

1986: Congress adds **Section 508** to the Rehabilitation Act of 1973, requiring electronic and information technologies to be accessible. This law is less than effective because it lacks a clear means of enforcing accessibility standards.

1990: **Americans with Disabilities Act (ADA)** "prohibits discrimination and guarantees that people with disabilities have the same opportunities as everyone else to participate in the mainstream of American life -- to enjoy employment opportunities, to purchase goods and services, and to participate in State and local government programs and services." (source: www.ada.gov/ada_intro.htm)

1998: Congress strengthens **Section 508** of the Rehabilitation Act of 1973. The revamped version of the law requires Federal agencies and other organizations that receive Federal funds to adhere

to strict accessibility standards. This version also includes more effective means of enforcing the standards.

2008: **ADA Amendments Act (ADAAA)** broadens the definition of "disability" and extends the ADA to groups that had originally not been adequately protected.

2010: Department of Justice announces that it is considering expanding the **ADA** to include the internet, particularly state and local websites and e-commerce sites. As of 2017, this expansion has not occurred.

2017: **Section 508** is "refreshed" to bring its technical specifications into closer alignment with international standards.

Digital Accessibility: The Legal Landscape

In the United States, the legal accessibility requirements for private companies are not as straightforward as one might assume. At the Federal level, two laws safeguard the rights of disabled Americans; however, neither law directly requires all private companies to make their websites and other digital products accessible.

Americans with Disabilities Act

The first of these laws, the Americans with Disabilities Act (ADA), became law in 1990 and requires that places of public accommodation meet certain accessibility standards. When this law was written, most people's relationship with technology was much different than it is today:

- The Internet was very limited and not widely known or available.
- Few people had personal computers, much less email addresses and smart phones.
- Websites and digital publications were virtually non-existent.

Given this, it is not surprising that the ADA does not mention the accessibility of websites, email, or any other digital products.

In 2008, in response to Supreme Court decisions that adopted a narrow definition of "disability," Congress amended the ADA to include a broad definition of the types of disabilities that were protected under the law. These amendments did not mention accessibility in relation to websites, email, or any digital products. Since then, the Department of Justice has periodically suggested that it was considering expanding the ADA to include access to technology, but this change has not yet occurred.

Although the ADA does not explicitly apply to websites and other digital products, there have been a number of lawsuits against companies accused of having inaccessible websites. These suits argue that websites are effectively "places of public accommodation" and are therefore covered under Title III of the ADA. Enough of these suits have been successful, including a highly publicized class action suit against Target, that many companies have chosen to make their websites accessible to avoid future legal action.

Section 508

Unlike the ADA, Section 508 of the Rehabilitation Act of 1973 explicitly deals with digital accessibility. Under this law, Federal agencies are required to make their electronic and information technology (EIT) accessible to people with disabilities. Federal agencies must abide by this law in the development, procurement, maintenance, and use of any EIT. Section 508 also provides strict standards to which EIT must adhere in order to be considered compliant.

While this law technically applies only to Federal agencies and organizations that receive funding from the Federal government, in practice many private companies find that they must comply with Section 508 standards in order to do business with the Federal government or with any organization that receives any of its funding from Federal sources. A private company that refuses to meet these accessibility standards would be cutting itself off from a significant number of potential customers.

Companies prove that their websites and digital products are "508 compliant" by completing a Voluntary Product Accessibility Template (VPAT). Project MUSE's most recent VPAT is available on our Accessibility Page.

Digital Accessibility: Moral Obligation

While the legal requirements can be somewhat ambiguous, the moral obligation for publishers to make their content as accessible as possible is clear: the fundamental purpose of academic publishing is to disseminate scholarship, and we can not accomplish this goal if we do not make every effort to reach the hundreds of millions of people worldwide who rely on assistive technology.

Print Disabilities

A print disability is any condition that limits a person's ability to interact with and extract meaning from a written document. This includes mobility challenges, such as the inability to hold a printed book or difficulty turning pages, as well as visual impairments, like blindness or low vision, which keep a person from being able to see the words of a digital book or journal on a screen.

Print disabilities also include learning differences, particularly dyslexia, which can make it difficult or impossible for a person to understand text even though they are physically able to see it.

A few simple accommodations can have a profound impact on how a person with a print disability experiences a digital publication:

- Text that can be increased by 200%, 300% or even 400% can make a difference between a piece being legible or illegible to someone with low vision.
- Embedded fonts that screen readers can read aloud give both blind users and dyslexic users access to the content.

- Marking text up to indicate reading order allows assistive devices to move through text in the most logical and useful order to maximize comprehension.
- Carefully selecting colors used on a webpage (for backgrounds, icons, symbols, etc.) can help people with low vision or color blindness use the page.
- Allowing users to override a site's default design in favor of their own personal style sheets, which can be tailored to meet their particular accessibility needs.

Hearing Disabilities

Why should publishers of academic books and journals have to do anything to make their content more accessible to someone who is deaf or hard of hearing? Are not written documents, by their very nature, accessible to these people?

Yes and no. Certainly, a hearing disability should not have any obvious effect on a person's ability to read the text of digital book or journal. However, one of the most exciting aspects of a digital publication is that it can contain a variety of supplements that are not necessarily based on text. For example, a literary magazine could preface each short story with a video of the author talking about the piece. Similarly, an anthropological study of traditional foods could contain videos of cooks preparing the dishes and talking about the ingredients.

The possibilities for adding value to a book or article in this way are almost limitless. Unfortunately, this added value will be largely unavailable to deaf and hard of hearing readers unless the videos are accompanied by closed captions and/or transcripts and unless the website has a video player installed that can use this captioning.

Part 2:
Web Content Accessibility Guidelines

WCAG 2.0: An Overview for Publishers

The Web Content Accessibility Guidelines (WCAG) 2.0 is product of W3C's Web Accessibility Initiative. The WCAG 2.0 is written by and for website and software developers (with input from others, including disability advocacy groups). Most of it is well beyond what editors, production managers, or authors need to know in order to do their jobs. However, having a basic knowledge of WCAG 2.0 can help anyone in publishing to understand what she can do to support her organization's digital accessibility goals.

History of the WCAG

The movement to establish web accessibility guidelines began in earnest in the mid-1990's after Tim Berners-Lee, considered the father of the worldwide web, mentioned its importance in a speech. Within a few years, various organizations had compiled over 30 different sets of guidelines, which were eventually compiled into a single document. This document became the basis for the first version of the WCAG in 1999. In 2008, this version was superseded by WCAG 2.0.

What is the WCAG 2.0?

WCAG 2.0 is the international standard for web accessibility. It consists of a highly detailed set of standards designed to make webpages and their content as accessible as possible to people with a variety of disabilities that negatively affect their ability to use websites. These conditions include visual, auditory, physical, speech, cognitive, and neurological disabilities and affect millions of people.

The WCAG 2.0 features 12 guidelines, which are organized around four fundamental principles. Each guideline has testable assessment criteria. Web content is assigned to one of three levels of compliance – A, AA, or AAA – based on how well it meets these criteria. (Note: Project MUSE meets all Level A standards and many Level AA standards. Our ongoing site redesign will bring us to complete Level AA compliance.)

WCAG 2.0 Principles

The guiding principles of WCAG 2.0 are often represented with four key words: perceivable, operable, understandable, and robust. What follows is a synopsis based on "WCAG 2 at a Glance;" a definitive discussion is available on the W3C page.

Perceivable – Content and user interface components should be available in a way that users can perceive. In other words, if a person is not able use one sense, the content must be available in a form that can be perceived using a different sense.

For example, a video recording must be accompanied by captions and/or a transcript for deaf and hard-of-hearing users. Similarly, non-text content (such as images and icons) must include text descriptions that can be read by the assistive technology employed by visually impaired users. Purely decorative non-text components (like dividers between sections of text) should be invisible to assistive technology.

Operable – Webpages must be designed so that all users can navigate them. For example, all functionality must be available from the keyboard for people who are unable to use a mouse. Users who need additional time must have the option of suspending features that automatically end a dormant session.

Understandable – Both information and user interface components should operate in logical ways that any user can understand. For example, templates and user tools should be consistent across pages so that a user knows what to expect. Similarly, users should be warned in advance if clicking on a tool (perhaps a filter) will change or limit the information on the screen. If a user makes a mistake, there should be a clear and easy way to fix it.

Robust – Web content must be designed to work with all manner of assistive technology, including current and future versions.

Making Content Perceivable:
The Role of Editors, Authors, and Production Managers

Most WCAG 2.0 standards are aimed at website and software developers and do not have a lot to do with the work of academic publishers (other than the publishers' IT departments, of course.) However, there is one WCAG principle that can only be accomplished with the cooperation of editors, authors, and production managers: Principle 1 – Perceivable, which requires that "information and user interface components must be presentable to users in ways that they can perceive."

Guideline 1.1 goes on to explain that all non-text content (i.e., images, tables, charts, etc.) must have text alternatives that can work with assistive technology. Similarly, Guideline 1.2 requires that captions and audio descriptions accompany audio- and video-recordings so that the content can be perceived by all users regardless of their ability to see or hear.

Project MUSE's website is designed to deliver text-based alternatives to non-text content. For example, thumbnails of book covers are tagged in such a way that a person using assistive technology will know the title and author of the book.

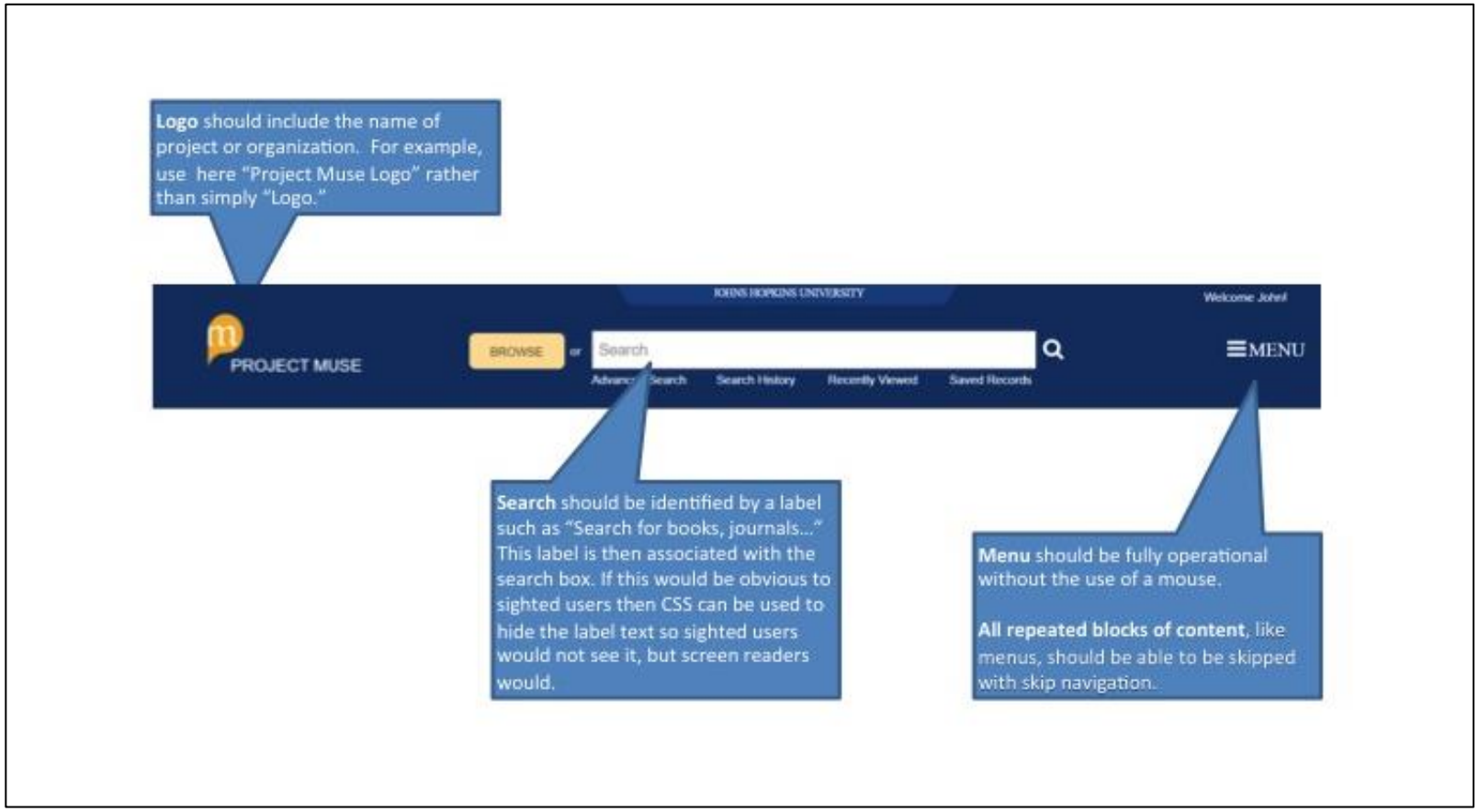
We rely entirely on our publishing partners to provide text-based alternatives for the non-text content that they submit to us. To cite just one example, Project MUSE's videoplayer is fully capable of displaying closed captioning *if* the publisher gives us captions along with a video. If

the publisher does not supply captions, we have no way to make a video accessible.

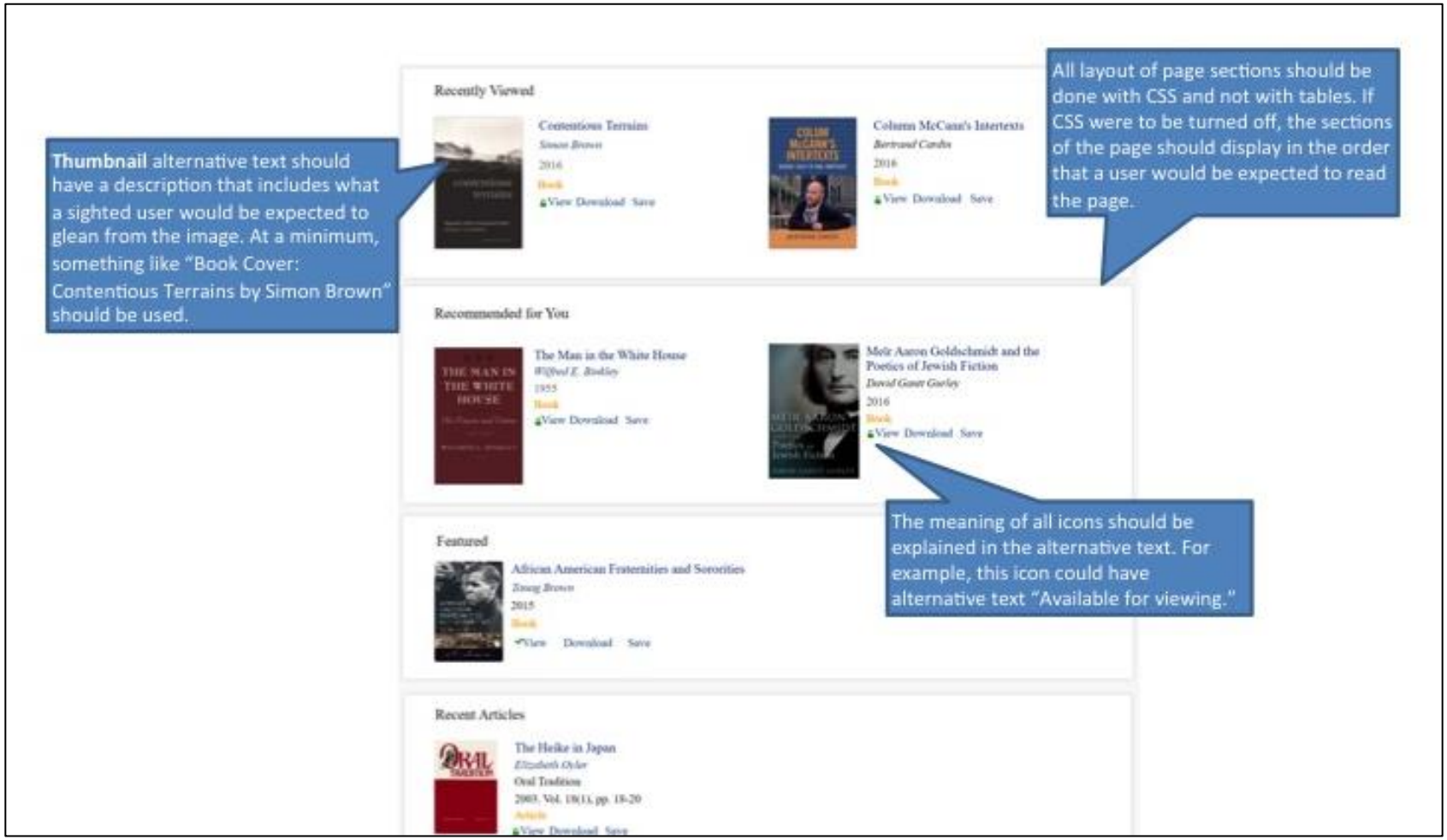
Project MUSE is committed to working with our publishing partners to help them develop efficient, cost-effective workflows for producing the text-based alternatives needed to make their content fully accessible to all users.

Part 3:
WCAG Principles Applied to MUSE Pages

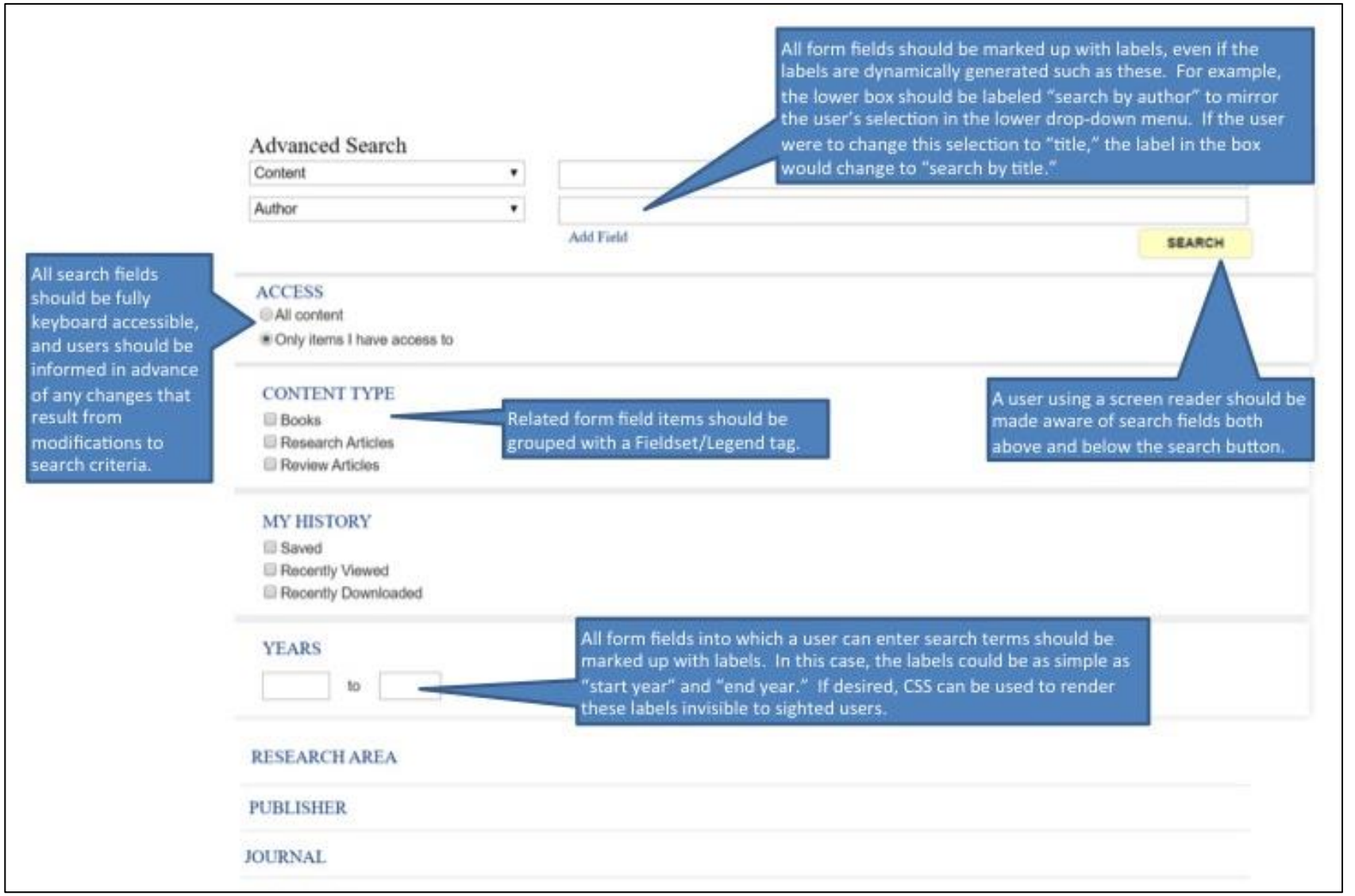
Site Header


Logo should include the name of project or organization. For example, use here "Project Muse Logo" rather than simply "Logo."
JOHNS HOPKINS UNIVERSITY
Welcome John!
PROJECT MUSE
BROWSE
or
Search
MENU
Search History
Recently Viewed
Saved Records
Search should be identified by a label such as "Search for books, journals..." This label is then associated with the search box. If this would be obvious to sighted users then CSS can be used to hide the label text so sighted users would not see it, but screen readers would.
Menu should be fully operational without the use of a mouse.
All repeated blocks of content, like menus, should be able to be skipped with skip navigation.

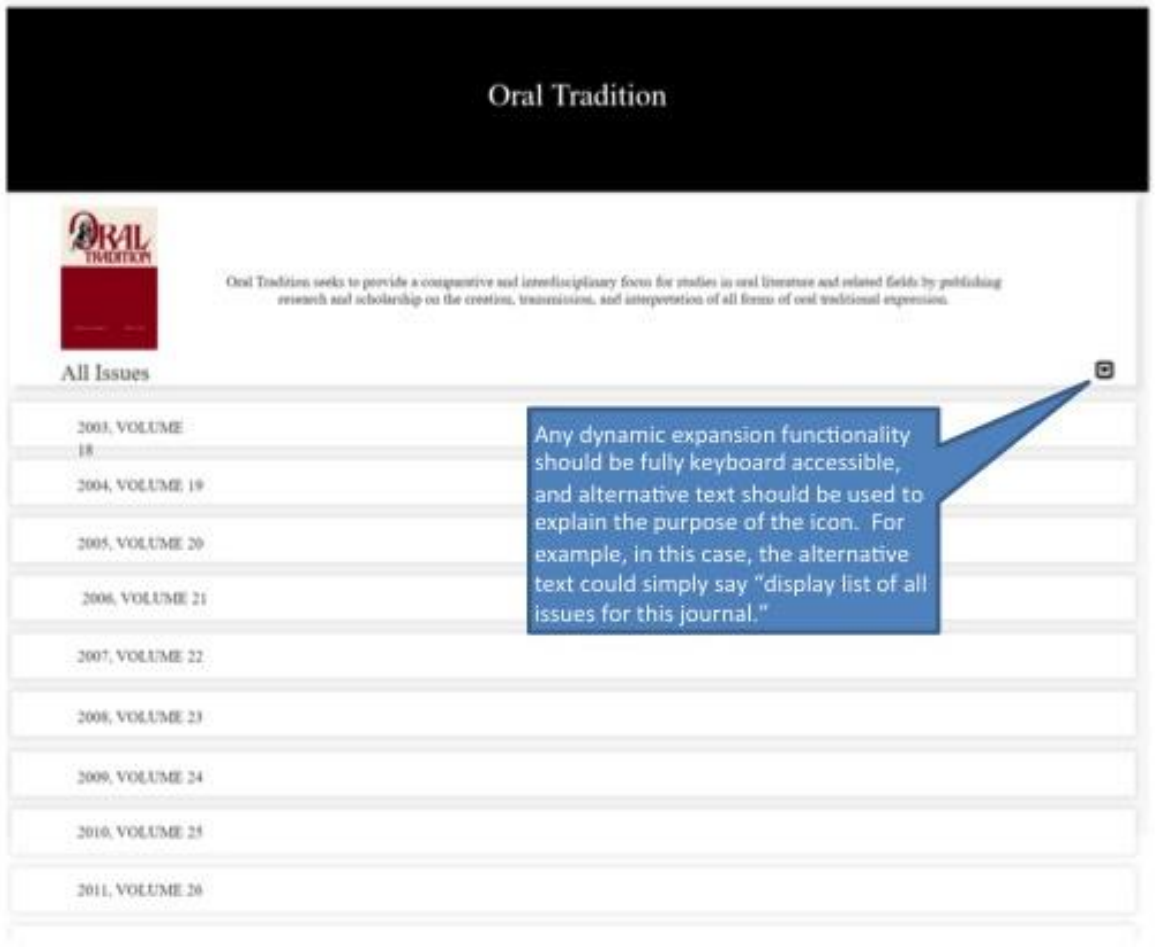
Home Page


Thumbnail alternative text should have a description that includes what a sighted user would be expected to glean from the image. At a minimum, something like "Book Cover: Contentious Terrains by Simon Brown" should be used.
All layout of page sections should be done with CSS and not with tables. If CSS were to be turned off, the sections of the page should display in the order that a user would be expected to read the page.
Recently Viewed
Contentious Terrains
Simon Brown
2016
Book
View Download Save
Column McCann's Intertexts
Bertrand Cardin
2016
Book
View Download Save
Recommended for You
The Man in the White House
Wilfred E. Binkley
1955
Book
View Download Save
Meïr Aaron Goldschmidt and the Poetics of Jewish Fiction
David Gantt Gurley
2016
Book
View Download Save
The meaning of all icons should be explained in the alternative text. For example, this icon could have alternative text "Available for viewing."
Featured
African American Fraternities and Sororities
Zoug Brown
2015
Book
View Download Save
Recent Articles
The Heike in Japan
Elizabeth Oyler
Oral Tradition
2003, Vol. 18(1), pp. 18-20
Article
View Download Save

Search / Advanced Search

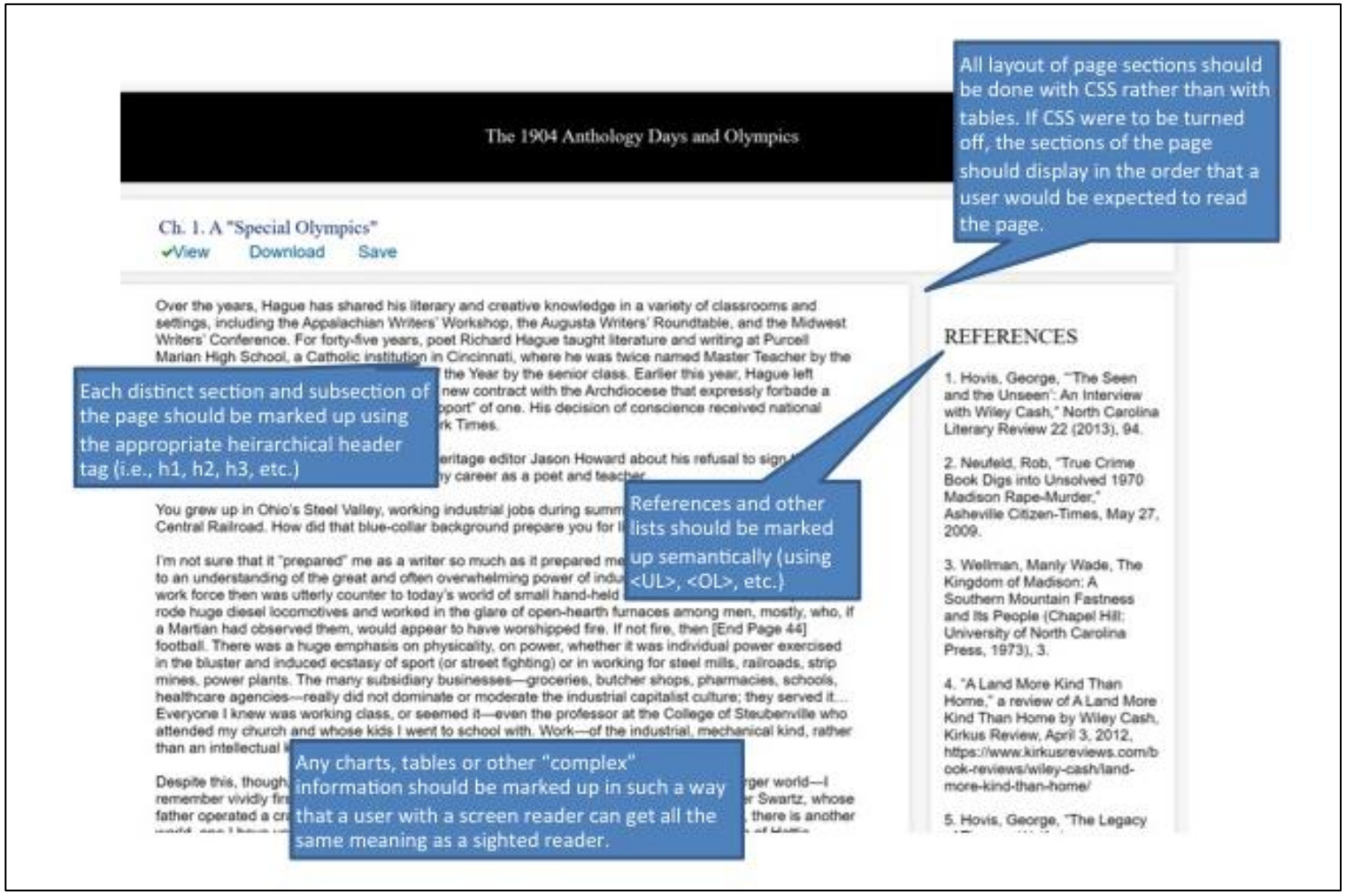


List of Issues



Single Journal Issue or Book

Individual Articles or Book Chapters



Appendix:
Applying and Validating WCAG 2.0 AA Standards

WCAG 2.0 Principle 1:
“Perceivable - Information and user interface components must be presentable to users in ways they can perceive.”

Guideline 1.1 Text Alternatives: “Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.”

Alternative text for images

- **Rule:** Any image that supplies meaning needs to have alternative text that is read by the screen reader.
  - The alternative text does not have to describe every nuance of the image but should make sure to describe whatever it is that a sighted person is supposed to get out of seeing that image.
- **How to check:**
  - Check the code and look for alt="TEXT HERE".
  - Use the WAT or WAVE toolbar to display alternative text for all images.
  - Use a screen reader and listen to what it says as it reads the image.
- **Additional Note:** Embedded multimedia (such as a video) needs to have alternative text too so that when a screen reader gets to the multimedia (before it starts playing), the screen reader announces what is embedded in the page.

Null alternative text for decorative images

- **Rule:** Any image that is only for visual decoration and supplies no meaning should have null alternative text. This will cause a screen reader to skip the image entirely.
- **Example:** An image of a blue line separates a book title from the book description. It would not add any value for a user to hear “blue line” from their screen reader, so the blue line is skipped.
- **How to check:**

  Check the code and look for alt=“” (there may be a space between the quotes).

  Use the WAT or WAVE toolbar to display alternative text for all images.

  Use a screen reader and check whether the screen reader skips the image.

Alternative text for complex images

- **Rule:** Any image that contains complex information needs alternative text that explains whatever key points a user is supposed to get from the image. This can be via the alt attribute, a longdesc attribute that contains a link or via a link to a separate page.
- **Example:** A journal article contains a bar graph. The key points that a user is supposed to get from the bar graph are contained in alternative text. If, however, there is too much to put into alternative text, a longdesc attribute may be put in and/or a link to a separate
- **How to check:**

  Check the code and look for alt="EXTENDED TEXT HERE" or longdesc = "URL HERE" or look for a link to another page near the image.

  Use the WAT or WAVE toolbar to display alternative text for all images.

  Use a screen reader and check whether the screen reader properly explains the image.

Frames / iFrames

- **Rule:** Any frames or iframes on a page should be labeled with a title attribute so that when a screen reader gets to the frame/iframe, it announces what is contained within.
- **Example:** A site pulls article text from a third-party source. Instead of pulling text directly into the page, however, the site uses an iframe and populates that iframe via a URL to that third-party source. The iframe could include a title attribute “Full text of article” so that when a screen reader gets to this point it tells the user what they’ll find once they enter this iframe.
- **How to check:**

  Check the code and look for <iframe title=“TITLE HERE”>.

  Use the WAT toolbar to display frame information.

  Use a screen reader and check whether the screen reader properly works with the iframe.

  Additional notes: While frames are rarely, if ever, used today, for accessibility purposes, it is ideal not to use iframes at all. Also be aware that users who have vision impairments (and don’t necessarily use a screen reader) may magnify a screen, so make sure that iframes still work when the screen is magnified at least 200% and ideally more).

Guideline 1.2 Time-based Media: "Provide alternatives for time-based media."

Audio and Video

- **Rule:**

  Prerecorded Audio that do not have any associated video needs to have a separate transcript that includes in written form all the words that are spoken and any background sounds that supply meaning and context.

  Prerecorded video needs to have synchronized captioning that goes along with the video. This also needs to include in written form all the words that are spoken and any background sounds that supply meaning and context.

- **Example:**

  – A site includes an embedded audio player at the top of the page. Immediately below the embedded audio player is the full text of the audio written out.

  – A site includes an embedded video player on the page. There is a button within the video player functionality that turns on captioning so that as words are spoken (or meaningful sounds are heard) the information appears in text simultaneously.

- **How to check:**

  Look at the page and see that the transcript or synchronized captioning is available.

Guideline 1.3 Adaptable:

“Create content that can be presented in different ways (for example simpler layout) without losing information or structure.”

Semantic Markup

- **Rule:** Proper HTML markup needs to be used to structure text. Headings with <H1>, <H2> and perhaps <H3> tags should be used to outline the major sections of the page.
  - Lists are marked up with <UL>, <OL> and <DL> and other tags such as <Strong> or <Blockquote> are used appropriately to markup text.
  - Tags are not used simply for the visual formatting they provide.
- **How to check:**
  - Check the code and make sure headings are marked up with <H> tags, etc.
  - Use the WAVE or WAT toolbar to display tags.
  - Use a screen reader and check whether the screen reader properly announces markup as expected.

Tables

- **Rule:** Tables should be used for tabular data; page content needs to be laid out only using CSS. Table header cells should be properly marked up as such. WebAIM has an article on how to code accessible tables.
- **Example:** A full-text article contains a table that shows data and is properly marked up so that a user using assistive technology will know that a data cell is associated with a row header and a column header.
- **How to check:**

  Check the code and look for proper table markup.

  Use the WAT or WAVE toolbar to display the table markup.

  Use a screen reader and confirm that it is possible to navigate through the table and know which row/column the cursor is on.

Reading/Navigation Order makes sense

- **Rule:** As a user using assistive technology (and often no mouse) goes through the page section by section, the presentation of sections should match the way that a user without assistive technology would be expected to go through the page with an order that makes sense.
  - This is true with both the order that page contents are read and the order that a user may tab through from link to link on the page.
  - This is also true with the navigation order of forms.

- **How to check:**
  - Tab through the links and see if the order makes sense.
  - Use a screen reader to read through all the text on the page and see if the order makes sense.

Sensory Characteristics

- **Rule:** Instructions should never request that a user do something based on a sensory characteristic (size, shape, color, location) or after a sound.
- **Example:**
  - Instructions would never say “Use the green button to activate the article search” since, for example, a blind or color-blind user would not know what button is green.
  - Instructions would never say “After the tone, select your publication” since a user who couldn’t hear would not know when the sound appeared.
- **How to check:** Review all page instructions and make sure that no sensory characteristics are represented.

Guideline 1.4 Distinguishable: "Make it easier for users to see and hear content including separating foreground from background."

Use of color

- **Rule:** Color cannot be used exclusively to provide meaning.
- **Example:** Red text alone cannot be used to distinguish errors from correct text. However red text that contains the word "Error" is okay because the word "Error" is not color dependent.
- **How to check:** Review pages for any situation where color is exclusively used to provide meaning.

Resize text

- **Rule:** Text should be able to me resized up to 200% and still be fully usable.
- **How to check:** Resize to 200% in the browser and make sure that the page is still usable.
- **Additional notes:** Even though 200% is required, try resizing even larger to help assure that the page does not break at higher magnifications.

Audio controls

- **Rule:** Any audio that automatically plays for more than 3 seconds should have an accessible mechanism to pause or stop the audio or there should be a way to control the page audio that is separate from the system audio.
- **Example:** An introductory promotion starts playing the moment someone gets to a page. A user using a screen reader is able to tab quickly to the promotion and gets accessible controls to stop the audio so that the user can hear the screen reader without interference.
- **How to check:** Make sure that there is a way to get to audio that is playing for more than 3 seconds and stop it. This should be able to be done without the use of a mouse.

Images of text

- **Rule:** Text should be presented as actual text, not as an image of text (unless the same visual presentation can't be made using text alone).
- **Example:** A publication demonstrates how cursive letters looked in the 18th century. Because this could not be done with regular type, it's okay to show the cursive letters as an image (with appropriate alternative text).
- **How to check:** Use the WAT or WAVE toolbar to identify images and make sure that nothing that should be regular type is identified as an image.

Color contrast

- **Rule:** Text should have a contrast ratio of 4.5:1 against the background color except for large text (18 point or 14 point bold) that can have a contrast ratio of 3:1.

- **How to check:**
  - Type the foreground and background color hex codes into the WebAIM Color Contrast Checker and confirm that it passes at the WCAG 2.0 AA level.

WCAG 2.0 Principle 2:
“Operable - User interface components and navigation must be operable.”

Guideline 2.1 Keyboard Accessible:
“Make all functionality available from a keyboard.”

Keyboard accessible

- **Rule:** Assistive technologies often do not use a mouse. Therefore, no matter how complex a web application may be, it must work fully with only a keyboard and without the use of a mouse. Additionally, keyboard traps (e.g., a user can tab into an area but not get out without performing a mouse action) must be avoided.
- **Example:** A user tabs into a search field but due to form validation, cannot tab out of the field until a search string is entered. The user decides not to do a search but since no mouse is available, the user is unable to leave the search field.
- **How to check:** Put the mouse away, and make sure that all functionality on the site can be used with only a keyboard.

Guideline 2.2 Enough Time:
"Provide users enough time to read and use content."

Timeouts

- **Rule:** If there is a time limit anywhere in the application, the user should be able to turn off or adjust the time limit to allow for more time.
- **Example:** A user is allowed to log in for a period of 1 hour. Towards the end of that time period, the system alerts the user with a message that indicates that the system is soon going to log the user out. The user then has the opportunity to select an option to allow for another hour.
- **How to check:** To assess system timeout rules, it is often necessary to talk with the development team and figure out what those rules are. Once rules are understood, they can be tested to make sure that they work as expected.
- **Additional notes:**
    - One way that users can be given more time is with a pop-up message. However, since this pop-up message only occurs in time-out situations, it is often not tested in an accessibility check.
    - Make sure that the pop-up message is tested with a screen reader to assure that it works as expected. That is, the screen reader focuses on this message when it occurs, and a user can use a keyboard to deactivate the message and/or allow for more time.

Automatically moving, blinking, scrolling, or updating text

- **Rule:** Any information that automatically moves, blinks, scrolls or updates for longer than 5 seconds and is presented in parallel with other page content should be able to be paused or stopped.
- **Example:** The home page of a publications site has a ticker feature that constantly shows users new and updated resources. This ticker feature has a button to pause or stop it so that, for example, a user who is unable to focus on other areas of the page because of this constant motion will now be able to do so.
- **How to check:** Look for any automatically moving, blinking, scrolling or updating text and make sure that it can be stopped using some keyboard-accessible means.

Guideline 2.3 Seizures:
“Do not design content in a way that is known to cause seizures.”

Avoid flashing

- **Rule:** Nothing on the page can flash more than three times per second.
- **Example:** The publication site hosts a small advertisement that flashes to get users’ attention. If the flashing is measured at more than three times per second then this flashing could risk causing a seizure in sensitive individuals.
- **How to check:** Look at the page content and make sure that nothing is flashing. If anything is identified to be flashing, measure the flash rate and make sure that it is not greater than three flashes per second.

Guideline 2.4 Navigable:
"Provide ways to help users navigate, find content, and determine where they are."

Page titles

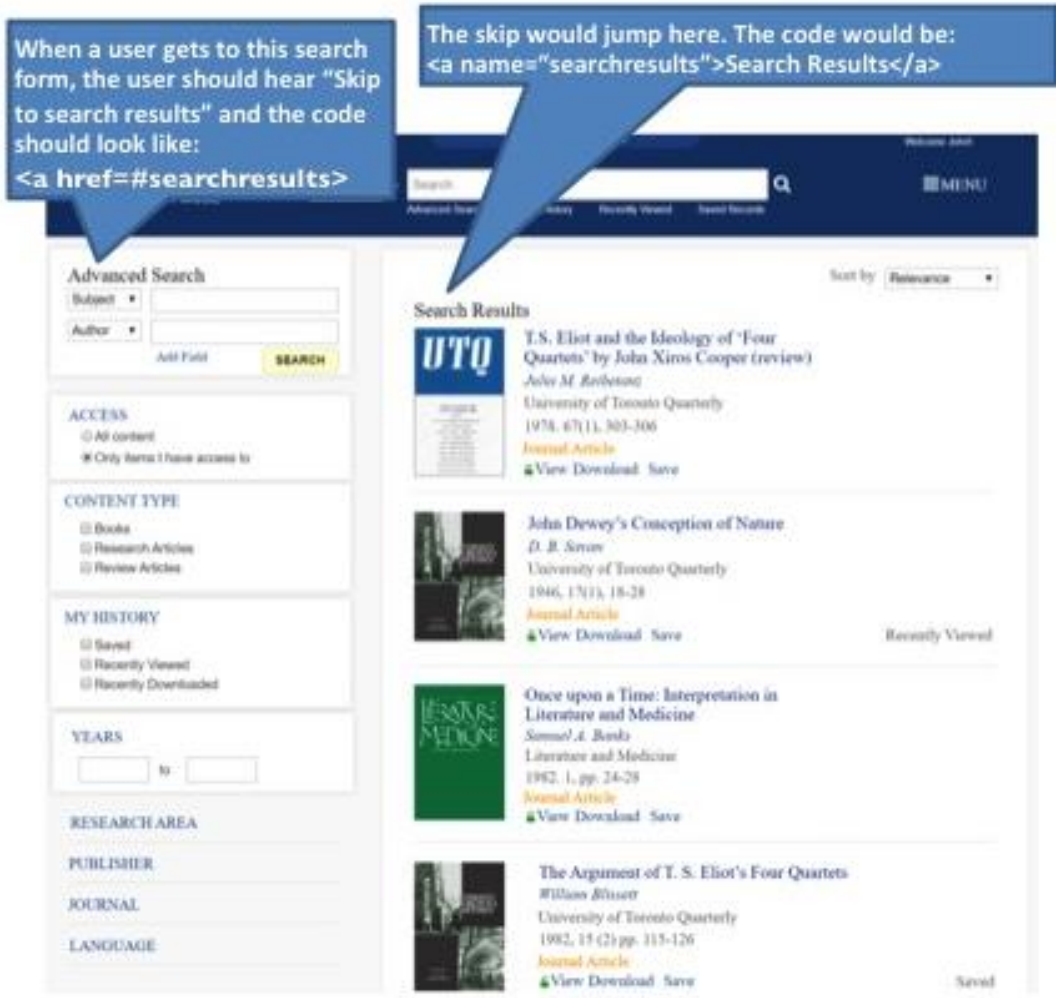
- **Rule:** All pages must have meaningful page titles within the <TITLE> tags. Page titles are often the first thing that assistive technology, such as a screen reader, will announce and should very briefly encapsulate the meaning of the page.

- **How to check:**
  - Look at the code to see what is contained within the <TITLE> tags.
  - Look in the browser tab at the page title that appears when a page comes up.

Skip navigation

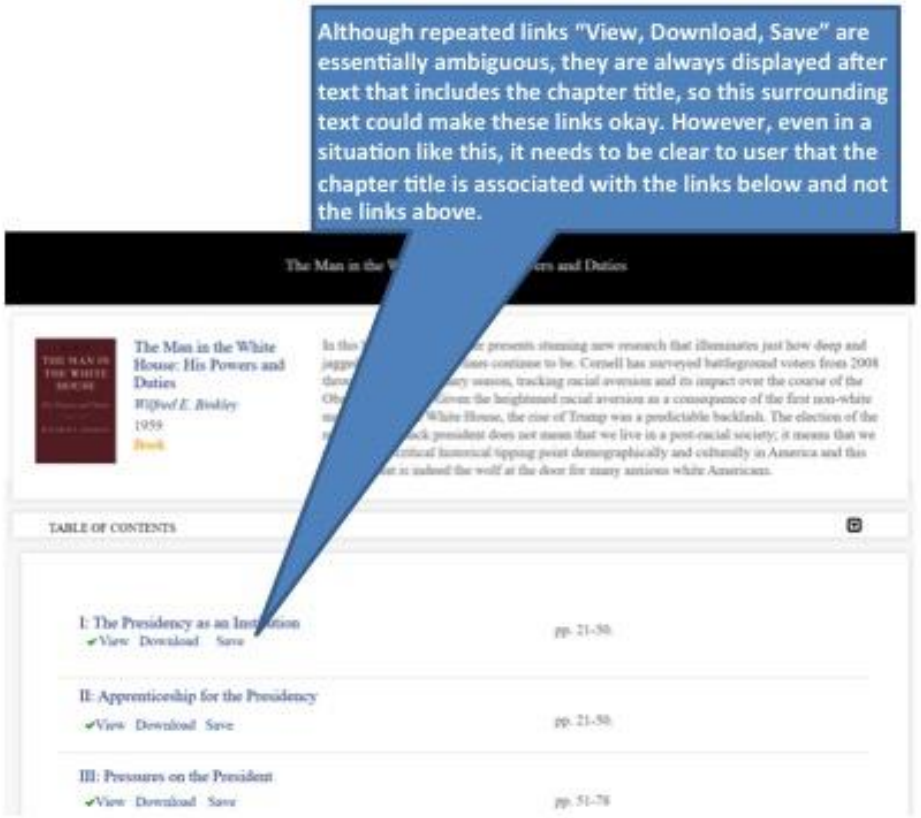
- **Rule:** A method should be provided to skip repeated blocks of content. This could mean consistent navigation or could be other types of repeated content such as, for example, a set of search fields that gets repeated across pages.

- **How to check:** Tab through a page to the point immediately before the repeated block of content and make sure that there is a link that skips to a point immediately after the repeated block of content.



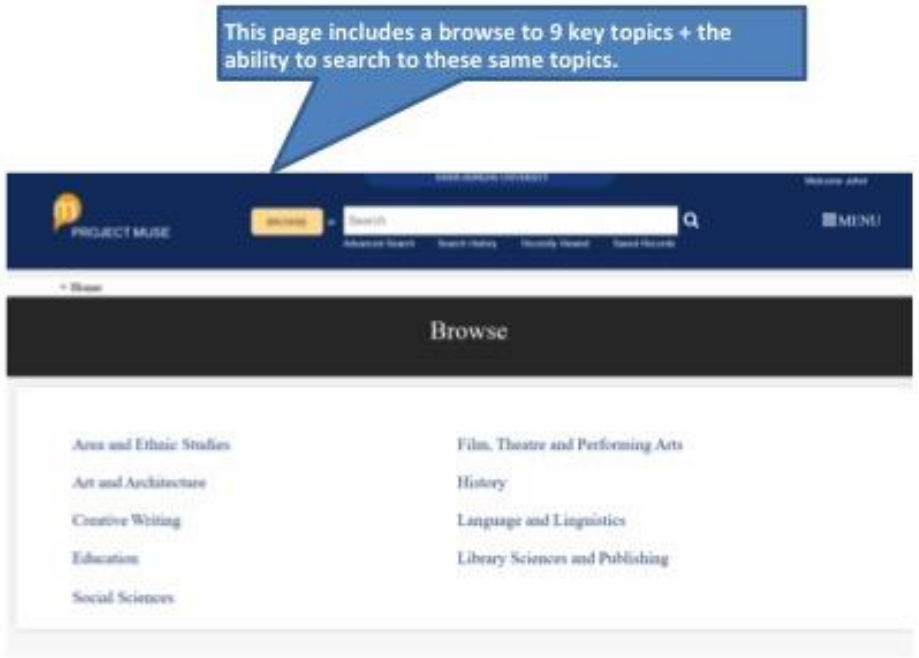
Link purpose

- **Rule:** The purpose of each link should be able to be determined from link text itself, or from surrounding text.
- **How to check:**
  - Tab through the links on the page and for each link, consider whether the link could be understood without anything beyond the text contained in the link or surrounding text.



Multiple ways of finding things

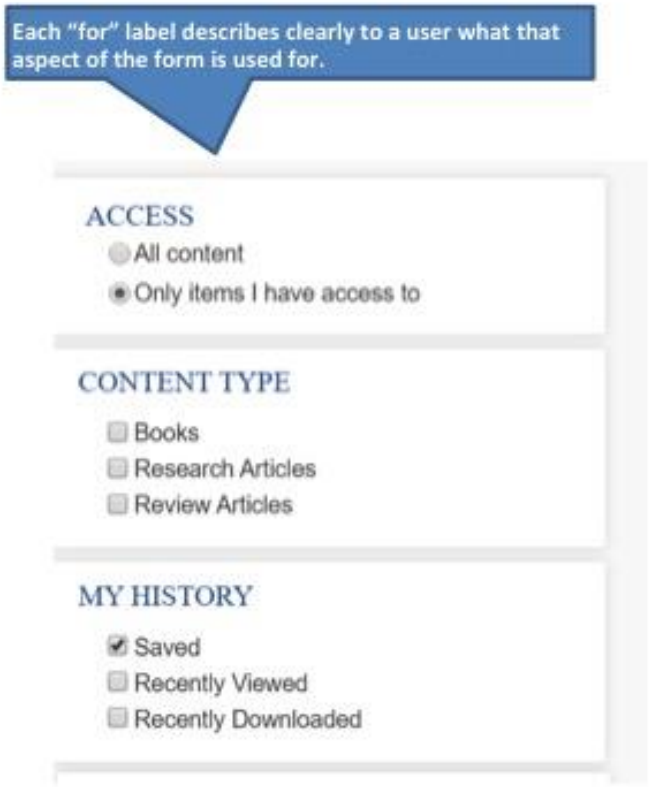
- **Rule:** There should be at least two ways to get to any page from any other page on the site. The two ways to get to a page could include any combination of the following:
  - Links in a main menu
  - Links in a footer
  - Related / within-page links
  - Search results
- **How to check:**
  - Review the site as a whole and make sure that there are at least two ways to get to every page.



Headings and labels must be meaningful

- **Rule:** Headings used on the page and labels used on forms and buttons must be meaningful enough to convey their purpose. Headings should not be generic nor should they be duplicated.

- **How to check:**
  - Look at the headings used throughout and make sure that they are properly descriptive.



Visual focus

- **Rule:** When a user is only using a keyboard and is tabbing through links and functionality, it should be clear exactly where the user is on each page. In the case of ordinary links, visual focus often means a box appears around the link when the cursor is on that link.

- **How to check:**
  - Tab through the links and functions on a page and make sure that the visual focus is always clear.

WCAG 2.0 Principle 3:
“Understandable - Information and the operation of user interface must be understandable.”

Guideline 3.1 Readable:
“Make text content readable and understandable.”

Language: Pages & parts of a page

- **Rule:** The language of the page needs to be identified. This is done, for example by simply adding a “lang” attribute to the <HTML> tag at the beginning of the page, such as <html lang=“en”> for English text.
- **Example:** A page is written in English, but includes the abstract of an article that was written in Spanish. The page includes <html lang=“en”> but the abstract portion includes where “es” means Spanish.
- **How to check:** Review the code to see if language tags are added properly.

Guideline 3.2 Predictable:
“Make Web pages appear and operate in predictable ways.”

Consistency of navigation items and functions

- **Rule:** Menu items should not change order when navigation is repeated across pages. Also, components with the same functionality should be labeled in the same way across pages.
- **Example:** Search should not be called “Search” on one page and “Article search” on another page if both search fields are searching the same content.
- **How to check:** Look at navigation and labels used across the site to make sure that things are consistent.

On focus / input

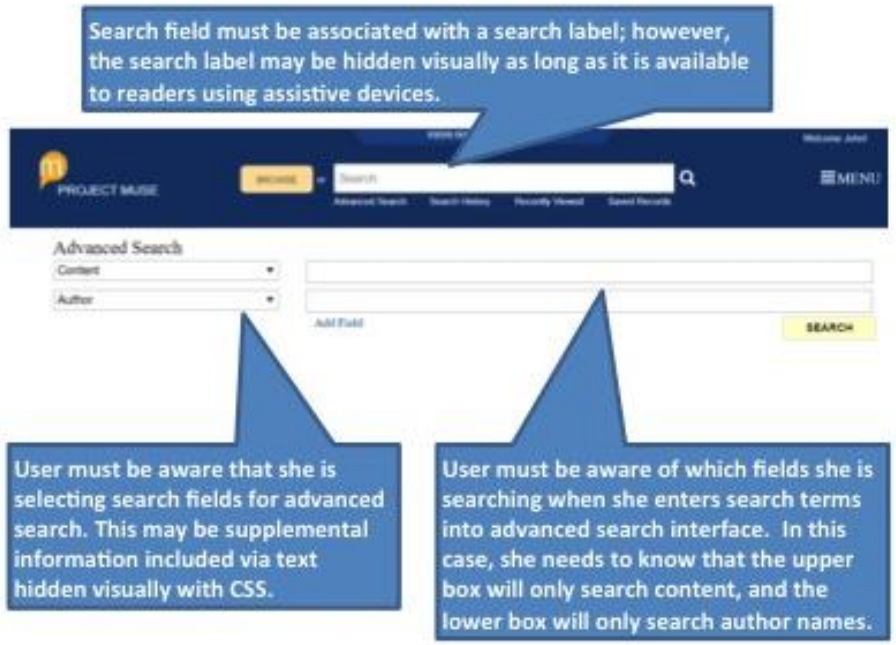
- **Rule:** When any component of the page receives focus it should not result in significant changes to the page. When a user inputs something into the system it should not automatically cause a significant change to the page without the user being made aware of that change.

- **How to check:**
  - Go through the site with a keyboard and check whether items receiving focus or user input cause any significant changes to the page.

Guideline 3.3 Input Assistance: “Help users avoid and correct mistakes.”

Forms, labels, instructions and validation info

- **Rule:** Forms need to be coded to be accessible. WebAIM provides detailed instructions on how to create accessible form labels and controls. Instructions must be provided in a way such that they are available to assistive technologies, and labels, fieldsets and legends must be used appropriately.
- **How to check:**
  - Check the code and confirm that the form is properly coded for accessibility.
  - Use the WAT or WAVE toolbar to show accessibility behind form controls.
  - Use a screen reader and try to complete the form without any other context beyond what can be heard through the screen reader.



Errors are identified accessibly

- **Rule:** Errors that are automatically identified (for example, upon form submission) should be described in text in an accessible way. If suggestions for correction are known, these suggestions should also identified (unless there are security concerns).
- **Example:** A user is asked to specify a password when creating a new account to gain access to a publication site. The system requires a number and a letter within the password, but the password that the user has entered contains only a letter. The user is informed in text that the password is not valid and is told that to create a valid password a number must be included as well.
- **How to check:** Make sure visually that the errors/suggestions appear, and then use a keyboard and screen reader to assure that this information is available to assistive technologies. Complete a form incorrectly using assistive technology and see what happens.
- **Additional notes:** Even if accessible, also make sure to put errors/suggestions in a location that a user would find it. So, for example, error information at the top or bottom of the screen may not be found if the user's cursor remains in the center of a form.

Legal or financial data

- **Rule:** Any form submissions that represent a legal commitment or a financial transaction should be reversible, are able to be fully checked for errors in an accessible way, and are able to be reviewed and if necessary corrected before being finalized.
- **Example:** A publication site requires a subscription fee. The subscription represents a legal commitment and the payment of the fee represents a financial commitment.
- **How to check:**
  - Check the code and make sure that forms are properly coded to be accessible.
  - Use the WAT or WAVE toolbar to check forms.
  - Use a keyboard and screen reader and check whether all errors and form data itself can be accessibly reviewed before submission.

WCAG 2.0 Principle 4:
"Robust - Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies."

Guideline 4.1 Compatible: "Maximize compatibility with current and future user agents, including assistive technologies."

Backend Requirements

- **Rule (for developers):** HTML errors should be avoided, when the page is tested with the W3C validator and names, roles and values should be be programmatically determined and work fully with assistive technologies.
- Example: Developers develop code properly throughout with an eye towards accessibility. Once the site is coded properly and is mostly compatible with screen readers and other assistive technologies, additional ARIA coding can be added to help enhance accessibility features.
- How to check: The easiest way to check proper coding is to test out the site using a screen reader. Keep in mind, however, that different screen readers have differing levels of ability to compensate for coding errors, Just because a known coding error doesn't cause problems in one screen reader does not mean that the site will be universally usable in all forms of assistive technology.

•