

High-level accessibility review – BTAA (Alexander Street Press - final version)

Primary Point of Contact

Denis Boudreau
Principal Web Accessibility Consultant

Deque Systems, Inc.

Web: www.deque.com

Email: denis.boudreau@deque.com

October 15th, 2017



High-level accessibility review – Alexander Street Press

TABLE OF CONTENTS

High-level accessibility review – Alexander Street Press	2
Alexander Street Press	3
Summary (top 3 problems for Alexander Street Press)	3
Accessibility findings	3
Project wide issues.....	3
1. Alexander Street Press landing page	4
2. Alexander Street Press simple search page	5
3. Alexander Street Press – Test a document	7
4. Alexander Street Press – Test a video.....	9
5. Alexander Street Press – Advanced search feature.....	11

Alexander Street Press

Summary (top 3 problems for Alexander Street Press)

While the overall level of accessibility implemented in this platform is surprisingly interesting, there are still some significant issues that will create problems for some users with disabilities. Beyond color contrast issues that were constantly brought up by our automated tests, the main accessibility barriers found with the Alexander Street Press platform were related to how well the site works with assistive technologies and for those who cannot operate a mouse. The top 3 issues found in this application were:

1. **Screen reader compatibility** – information is not always reliably conveyed to users, which can create confusion. A lot of effort has been put into providing helpful messages and indications, but this has not been done in a consistent way throughout.
2. **Imperfect use of ARIA** – the platform implements WAI-ARIA technology to accomplish a lot of great things, but there are still problems that will potentially make it difficult for screen reader users – some of it is related to misuse of ARIA, some of it is less than ideal implementations.
3. **Video accessibility** – The media player shows issues when it comes to accessibility, both from the perspective of keyboard navigation and screen reader compatibility. There is an issue with synchronicity between the captions and the transcript that gets in the way of screen readers.

Accessibility findings

Project wide issues

The issues presented in this section were identified in multiple pages, and are recorded here to avoid repetition. These are applicable to each screen.

Automated findings using Axe

- **SC 1.4.3** – Color contrast ratios are insufficient on multiple objects in the page.
- **SC 4.1.1** – ID attribute values used must be unique. ID value “edit-search-api-views-fulltext-autocomplete-aria-live” is referred to twice.
- **SC 4.1.2** – Some of the WAI-ARIA roles used in the code base have not been assigned a valid value. The “section” role is an abstract role that should not be used within content.

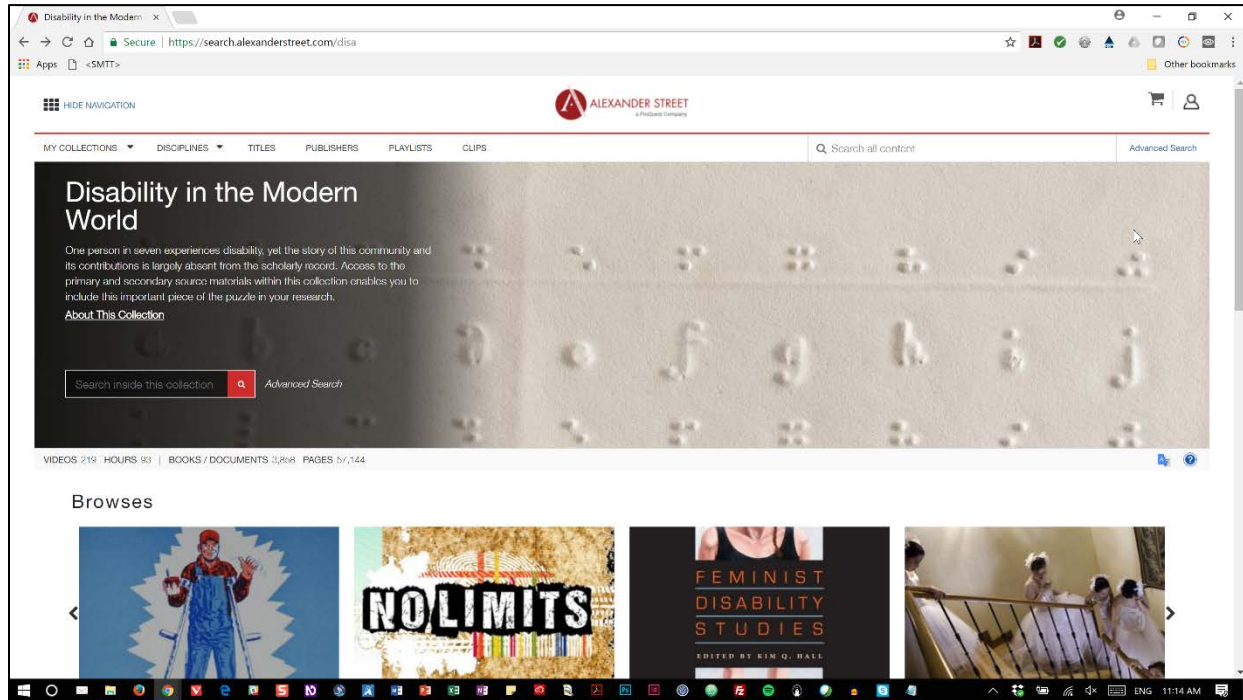
Additional manual findings using NVDA screen reader

- **SC 3.3.2** - The Search field relies on a placeholder attribute to give it a text label. That label disappears as soon as content is typed into the field. This may cause issues for some users with cognitive disabilities.
- **SC 4.1.2** - The state (collapsed or expanded) of the main navigation is not announced as screen reader users toggle the section.

1. Alexander Street Press landing page

Source: <https://search.alexanderstreet.com/disa>

Use case: Test initial interface/landing page to ensure menus, search box, images, icons, categories below search box, etc. are accessible.



Automated findings using Axe

- **SC 1.1.1** – The background image of braille used at the top of the page must either have a descriptive alt text value, or be assigned a role of presentation.
- **SC 1.4.3** – Color contrast ratios are insufficient on multiple objects in the page.

Additional manual findings using NVDA screen reader

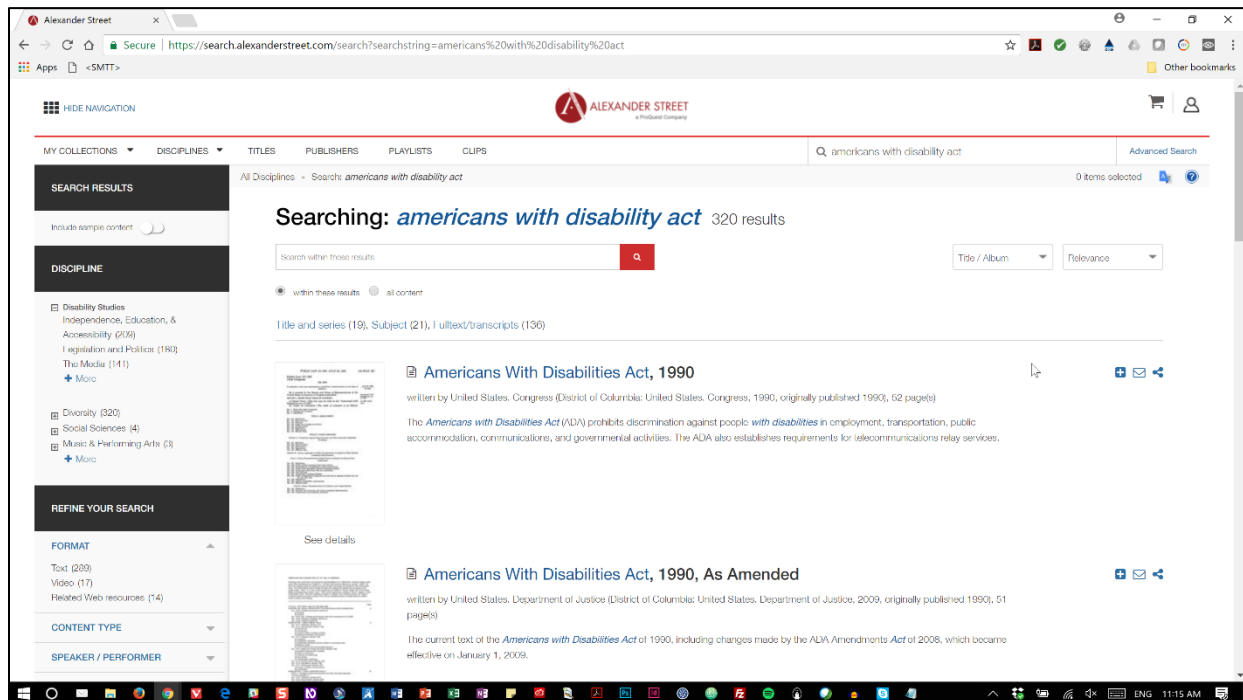
- **SC 2.1.1** - The previous and next arrow controls in the carousel cannot be reached using the keyboard alone.
- **SC 2.1.1** - The language selection button cannot be triggered using only the keyboard.
- **SC 2.4.3** - As focus is moved away from the language modal, it is not returned to the initiating button, but rather, is randomly sent at the bottom of the page.
- **SC 2.4.7** - The focus indicator is not clearly visible as one tabs through the page. Keyboard users lose sight of the focus when it reaches certain elements in the page, like the carousel.
- **SC 3.3.2** - The search field for searching inside a collection loses its text label when content is typed into the field.

2. Alexander Street Press simple search page

Source:

<https://search.alexanderstreet.com/search?searchstring=americans%20with%20disability%20act>

Use case: From initial interface, enter a simple search for: “Americans with disability act” and select “Search”. Test search results page, including: search within these results searchbox, sort by dropdowns (top right), adding results to the folder (plus sign below each result), menus on left (Search results, disability studies, refine your search). Test refine search options on left-hand side of search results page. Can these various search limit categories be expanded with just a keyboard? Are the menus/items within menus readable by a screen reader?



Automated findings using Axe

- **SC 1.3.1** – Radio buttons used for the Search feature are not part of a grouping.
- **SC 4.1.1** – ID attribute values used must be unique. Multiple ID values are used more than once.
- **SC 4.1.2** – Elements must only use allowed attributes, according to their roles. Multiple objects in the page rely on ARIA attributes that cannot be used in context.
- **SC 4.1.2** – Some expected ARIA attributes are not being used in context of the slider controls, where they are expected to be present (aria-valuemax and aria-valuemin).
- **SC 4.1.2** - Non-conforming attributes are used to define the slider controls (aria-minvalue and aria-maxvalue).
- **SC 4.1.2** – Multiple invalid aria-labeledby values are being used and are referencing ID values that are not present in the page.

Additional manual findings using NVDA screen reader

- **SC 1.3.1** – Heading levels are being skipped in the structure of the filters on the left. Headings jump from H3 to H5.

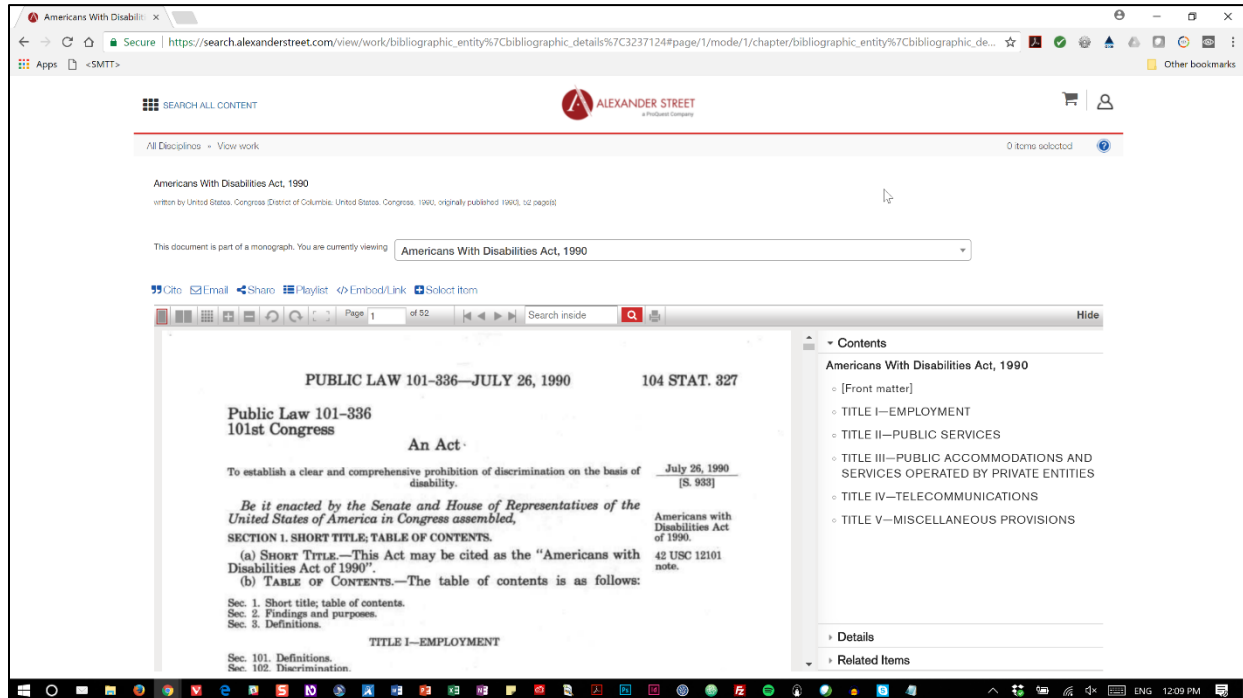
- **SC 1.4.1** – Links cannot be distinguished from their surrounding text in a way that does not rely on color alone.
- **SC 2.1.1** – The keyboard focus does not remain within the “Details” modal, but goes back to the content underneath while the modals remains in view.
- **SC 2.4.3** – Keyboard focus is lost once the “Include sample content” control is toggled or modal windows are dismissed. It is sent back at the very top of the page, instead of being repositioned on the control itself. The resulting change of state is not conveyed to assistive technologies either.
- **SC 2.4.4** – The “More+” links displayed in the left sidebar are not meaningful, even within their immediate context. Non-sighted users are likely to struggle with understanding what the links are for when these links are extracted from their context.
- **SC 2.4.7** – Keyboard focus is no longer visible when users reach the slider controls under “Date published / release” and “Date written / recorded”. Users might easily miss that those controls can be interacted with.
- **SC 4.1.2** – The state (collapsed or expanded) of up or down arrows in the left sidebar is not conveyed to users of assistive technologies.
- **SC 4.1.2** – Information is completely conveyed to screen reader users when reaching the graphs under “Date published / release” and “Date written / recorded”. Non-sighted users are very unlikely to understand what these objects are, much less understand their purpose. Instructions would be required, as well as an understanding of the role played by the sliders.
- **SC 4.1.2** – Dropdown menu for sorting in top right corner are not conveyed as such to screen reader users. No indications are provided to help users contextualize what the constructs are for. Users don’t hear about the various options either.
- **SC 4.1.2** – Embed modal forces NVDA into forms mode, making it impossible to explore the modal’s content. Only the active elements can be traversed, and the context for this content is lost to non-sighted users as a result. Assign a document role to the direct child of the modal to address this problem.
- **SC 4.1.2** – The “Remove/Add document feature is not clear, even within context. Users will be left wondering Adding or removing document to what, exactly? The labels need to be made a little more meaningful.

3. Alexander Street Press – Test a document

Source:

https://search.alexanderstreet.com/view/work/bibliographic_entity|bibliographic_details|3237124

Use case: Test a document for accessibility.



Automated findings using Axe

- **SC 4.1.2** – Current use of the ARIA combobox role is invalid, as the widget does not also leverage the use of the listbox and or textbox children roles.

Additional manual findings using NVDA screen reader

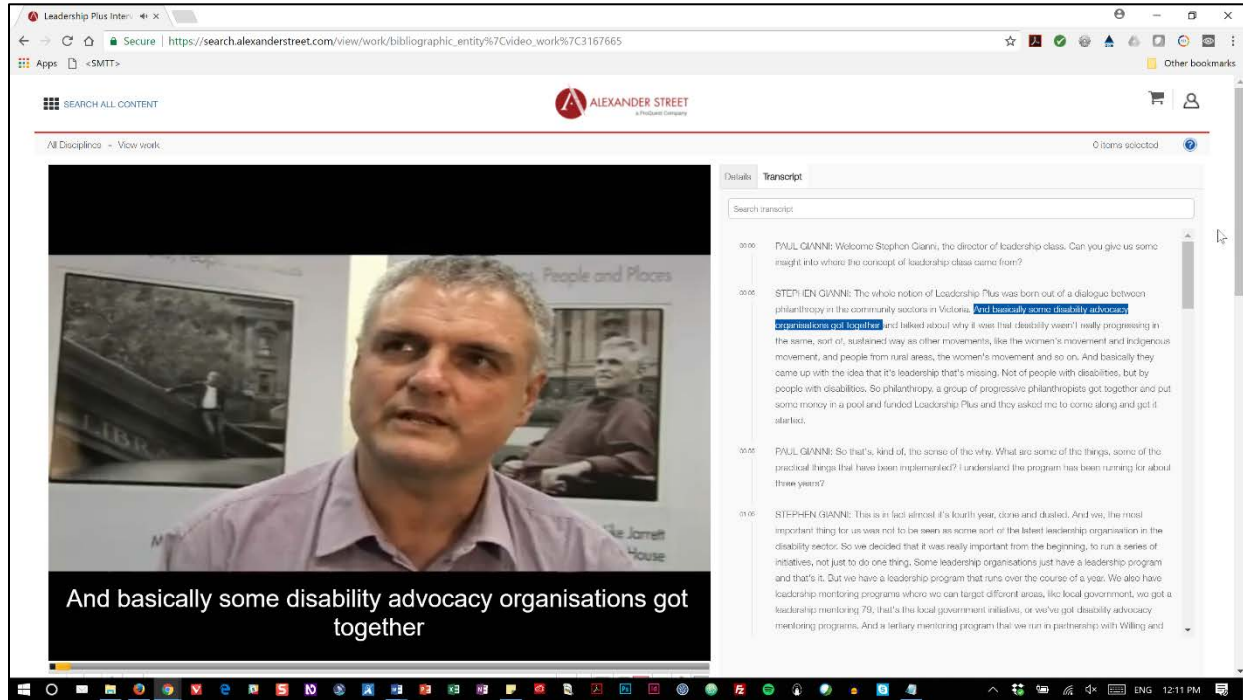
- **SC 1.3.1** – While the screen reader can have access to the content inside the viewer as it is turned into text using optical character recognition, there are issues with the document structure. The entire content is rendered as plain text, with no structure such as headings or lists.
- **SC 1.3.2** – While the screen reader can have access to the content inside the viewer as it is turned into text using optical character recognition, there are issues with the order in which the content is being read. Text is read line by line, even if it means jumping to other columns, instead of being read in the order in which the author defined it.
- **SC 2.1.1** – Focus is not set on the tabs on the right of the document viewer. The expanded tab has a tabindex attribute set to -1, forcing it into the tabbing order, but the other two are not. Those controls are only usable with a mouse.
- **SC 2.1.1** – Keyboard focus is not set on the Document viewer at all, preventing anyone not using a mouse from scrolling down the page. The control bar allows to jump to different pages, but only the top part is visible in the viewport. The bottom part of each page is lost to sighted keyboard users.

- **SC 2.4.3** – Focus is not managed when getting out of full screen mode. Keyboard focus is reset to the top of the page, instead of being sent back to the full screen button in the toolbar. Disability Media Australia,
- **SC 2.4.3** – Focus is not sent to new toolbar when the Print button is triggered. The new toolbar is not announced either and non-sighted screen reader users will struggle with understanding if anything happened as a result of triggering the button. Focus should be sent to the new toolbar.
- **SC 4.1.2** – The options found in the dropdown menu at the top for the monograph parts are all rendered as “blank” instead of conveying their respective values.
- **SC 4.1.2** – Embed modal forces NVDA into forms mode, making it impossible to explore the modal’s content. Only the active elements can be traversed, and the context for this content is lost to non-sighted users as a result. Assign a document role to the direct child of the modal to address this problem.
- **SC 4.1.2** – The accordion displayed on the right for the sections does not work properly with a screen reader. Beyond issues already mentioned with the keyboard, the controls are difficult to reach and sometimes are not conveyed at all. Consider implementing accordions following the WAI-ARIA design patterns, to provide a user interface component that will be easier to understand and make sense of: <https://www.w3.org/TR/wai-aria-practices-1.1/#accordion>.
- **SC 4.1.2** – The page often became unresponsive as a result of thoroughly testing it with a screen reader. This did not happen when testing the page without assistive technology, but happened on numerous occasions when NVDA was running.

4. Alexander Street Press – Test a video

Source: https://search.alexanderstreet.com/view/work/bibliographic_entity|video_work|316766

Use case: Test a video for accessibility.



Automated findings using Axe

- **SC 1.2.5** – Video must have an audiodescription track.
- **SC 2.4.1** – Iframes must be described with a meaningful title attribute value. The “session reset” iframe does not have such description.
- **SC 3.1.1** – The default language of the page must be defined, so assistive technologies are made aware of the language in which the document is written in. This default language value in this page is not defined.
- **SC 4.1.2** – The Make Clip button in the media player refers to an invalid ARIA attribute value for the aria-controls property.

Additional manual findings using NVDA screen reader

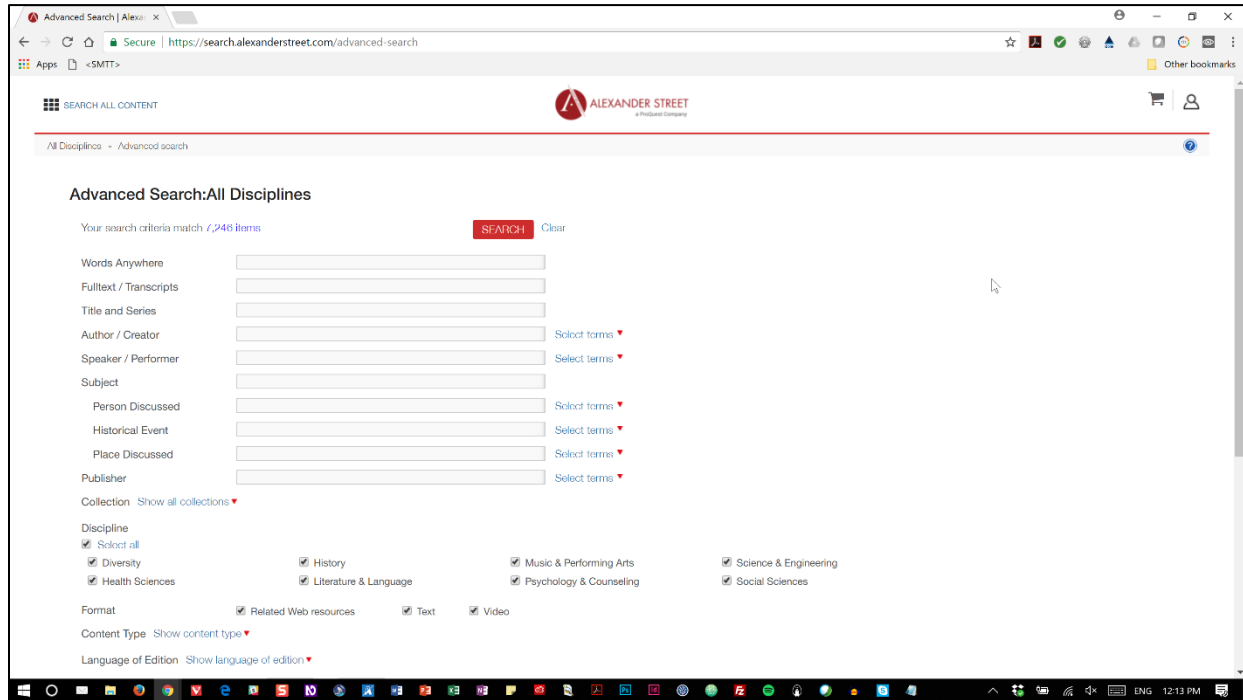
- **SC 1.3.1** – Elements that are made to look like headings in the Details panel are not marked up as such.
- **SC 2.1.1** – The star rating widget can only be operated with the use of a mouse, and cannot be controlled using only the keyboard, while each star can individually be tabbed to.
- **SC 2.1.1** – Part of the video controls do not appear to be keyboard triggerable: visual table of contents, change layout, captioning, quality levels and full screen mode do not appear to be usable with the keyboard only.
- **SC 2.1.1** – Once the title image for a video is triggered, the thumbnail for the video comes into view, but focus is not maintained into it. Therefore, it remains persistent, and keyboard-only users have no way to get rid of it, unless they can still perceive the outline when focus is sent back to the image.

- **SC 2.1.1** – Tab stops are forced on objects in the page that are not actionable (content in the Details tab panel). Consider removing the tabindex attributes from this content, and make the panel screen reader compatible instead by assigning a document role to the direct descendant of the tabpanel container.
- **SC 2.4.3** – Keyboard focus is lost once the modal prompted for controls requiring to be logged in is displayed. It is sent back at the very top of the page, instead of being repositioned on the originating control itself.
- **SC 2.4.7** - The focus indicator is not clearly visible as one tabs through the video controls. Keyboard users lose sight of the focus when it reaches certain controls, making it very difficult to use the player without the use of a mouse.
- **SC 4.1.2** – The screen reader does not announce the video object as focus is set to it, and tries to read caption right away, but freezes once the captions change. The current implementation does not allow for dynamic updates in the way the captions are conveyed.
- **SC 4.1.2** – Part of the video controls do not appear to be conveyed to screen readers: visual table of contents, change layout, captioning, quality levels and full screen mode do not appear to be compatible with screen readers.
- **SC 4.1.2** – Once the title image for a video is triggered, the thumbnail for the video comes into view, but focus is not set to it and screen reader users are not made aware of what happened.
- **SC 4.1.2** – The “Select Item” link item in the menu bar does not convey its current state to assistive technologies. Users can visually identify a change in the interface, but that change is not conveyed to non-sighted users in a way they can understand.
- **SC 4.1.2** – Embed modal forces NVDA into forms mode, making it impossible to explore the modal’s content. Only the active elements can be traversed, and the context for this content is lost to non-sighted users as a result. Assign a document role to the direct child of the modal to address this problem.
- **SC 4.1.2** – Screen reader users cannot traverse the transcript tabpanel to read the content of the transcript at their own leisure. Focus is set to the container, the interaction within the container is hijacked by the scrollbar and no content is read out loud.
- **SC 4.1.2** – The tabpanel implement poses problems for screen reader users. Since both tabs are presented before the content is traversed, this means users are likely to even forget that they are in a tabbed interface and go straight back to the top of the page. Consider implementing tab panels following the WAI-ARIA design patterns, to provide a user interface component that will be easier to understand and make sense of: <https://www.w3.org/TR/wai-aria-practices-1.1/#tabpanel>.

5. Alexander Street Press – Advanced search feature

Source: <https://search.alexanderstreet.com/advanced-search>

Use case: Test “Advanced Search” page for accessibility (link found right of search box from initial landing page) - no need to do an actual search/analyze results page since that’s covered above.



Automated findings using Axe

- **SC 1.1.1** – The down arrow images used next to the Select Terms controls have no alt text provided. They must either have a descriptive alt text, or be assigned a role of presentation.
- **SC 4.1.1** – ID attribute values used must be unique. Multiple ID values are used more than once.
- **SC 4.1.2** – ARIA attributes are being used in a context in which they are not allowed. Anchor elements rely on the aria-pressed attribute.

Additional manual findings using NVDA screen reader

- **SC 2.1.1** – The Clear (selected list items) link is not keyboard accessible.
- **SC 1.3.2** – While the list of checkboxes for any Search filter is ordered alphabetically per column, tabbing through the list of checkboxes does not follow the alphabetic order, but rather follows the linear table layout (row by row, cell by cell). This might prove somewhat challenging for some people with cognitive disabilities.