Accessiblü

Managed Digital Accessibility Ops

Accessibility Evaluation Report:

Bloomsbury Collections

February 18, 2026

Conducted by: Accessiblü, LLC

For: Library Accessibility Alliance (LAA)

Prepared by:
Accessiblü, LLC
882 Pompton Ave, STE A-2
Cedar Grove, NJ 07009
hey@accessiblu.com

Primary Contact:
Jeff Rodgers, MS Ed
Director of Digital Accessibility
Accessiblü
jeff@accessiblu.com

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Table of Contents

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Summary of Accessibility Findings

Accessiblü conducted a high-level accessibility evaluation of the Bloomsbury Collections platform (bloomsburycollections.com), a digital academic library providing access to scholarly books, reference works, and eBook content across the humanities, social sciences, and arts. The evaluation was performed on behalf of the Library Accessibility Alliance (LAA) to assess the platform's usability for users with disabilities.

Testing was conducted using the JAWS 2025 screen reader on a Windows 11 desktop with Google Chrome, supplemented by NVDA, keyboard-only navigation, and automated scanning via the axe DevTools browser extension for conformance against select WCAG 2.2 AA success criteria. Four pages were evaluated: the Home page, the Search Results page, the Advanced Search page, and an individual Book/Product page (monograph detail view).

The platform demonstrates a solid structural foundation with clearly defined landmark regions, a functioning skip link, and some well-labeled interactive controls. The main navigation and search form convey basic state information, and the site's color scheme provides good contrast in most areas. These are encouraging signs that the development team has given thought to accessibility.

At the same time, the evaluation identified several areas of opportunity that, when addressed, will meaningfully reduce friction for users relying on assistive technologies. The most significant patterns observed involve the home page carousel components, filter panel keyboard accessibility on the search results page, modal dialog focus management across multiple pages, and inconsistent ARIA role and property usage throughout the interface. Addressing these opportunities will improve the experience for blind and low-vision users, keyboard-only users, and screen reader users across the platform.

Key Findings

The evaluation identified accessibility opportunities that may create challenges for users relying on assistive technologies. A number of the carousel features on the home page lack the accessible names, structure, and keyboard controls needed to make them usable for screen reader users. Across multiple pages, interactive elements including navigation menus, filter panels, modal dialogs, and custom widgets use ARIA attributes inconsistently or incompletely, which can cause confusion for users navigating via keyboard and screen reader. Form controls on the Advanced Search page and the Book page lack proper programmatic labels, preventing screen reader users from understanding the purpose of those fields. None of these concerns represent insurmountable challenges, and the platform's overall architecture means targeted code-level fixes can address most issues systematically.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Top 3 Issues Identified

1. Carousel Components Inaccessible to Screen Reader and Keyboard Users

- Both featured book carousels on the Home page lack accessible names, proper ARIA listbox structure, and fully functional keyboard controls. The auto-advancing carousel has no accessible landmark or region label, leaving screen reader users unable to identify they have entered a carousel widget. Navigation controls (previous, next) for the first carousel were not reachable via keyboard during testing, and the position indicator buttons are labeled only as "cell indicator 0" and "cell indicator 1," conveying no meaningful information.
- Impact: Blind and low-vision users, keyboard-only users.
- WCAG Success Criteria: 1.1.1 Non-text Content (Level A), 1.3.1 Info and Relationships (Level A), 2.1.1 Keyboard (Level A), 4.1.2 Name, Role, Value (Level A).

2. Keyboard Inaccessibility of Filter Controls and Search Results Elements

- On the Search Results page, multiple interactive filter elements within the refine panel are not reachable or operable by keyboard alone. The filter accordion panels use an invalid ARIA attribute (aria-multiselectable) on the panel-group container, and several elements including sort dropdowns lack accessible names. A critical pattern is the complete keyboard-inaccessibility of numerous interactive elements throughout the search results interface.
- Impact: Users with motor disabilities, blind and low-vision users relying on keyboard navigation.
- WCAG Success Criteria: 2.1.1 Keyboard (Level A), 2.4.7 Focus Visible (Level AA), 4.1.2 Name, Role, Value (Level A).

3. Modal Dialogs Do Not Announce or Manage Focus Properly

- On the Book/Product page, the Share, Print, and Citation dialogs activate visually but do not announce themselves as modal dialogs to the screen reader. Focus does not move into the dialog upon activation, and on close, focus is returned to the top of the page rather than the triggering element. Additionally, on the Advanced Search page, the help dialog allows screen readers to read content outside the dialog while it is open.
- Impact: Blind and low-vision users, neurodiverse users relying on predictable interaction patterns.
- WCAG Success Criteria: 1.3.1 Info and Relationships (Level A), 2.4.3 Focus Order (Level A), 4.1.2 Name, Role, Value (Level A).

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Disabilities Impacted

Blind and Low-Vision Users

- Issues: Missing or inadequate alternative text on carousel book cover images. Carousel widgets lack accessible names and proper ARIA landmark structure. Navigation submenus include redundant link elements that interrupt the button-based pattern. Modal dialogs do not announce their role. The Download button on the Book page has aria-hidden="true" while remaining focusable, making it invisible to screen readers. The pagination input field on the Book page has no programmatic label.
- Impact: Screen reader users navigating via headings, landmarks, or tab order encounter gaps in the information architecture that require extra effort to work around, or in some cases prevent task completion entirely without sighted assistance.

Users with Motor Disabilities

- Issues: Keyboard-inaccessible interactive controls on the Search Results page, including filter elements and content widgets. Navigation dropdown menus are not fully operable by keyboard (spacebar does not activate buttons consistently). Carousel navigation buttons for the first home page carousel are not reachable by keyboard. Focus moves to the top of the page after closing modals, requiring users to re-navigate lengthy content to return to their place.
- Impact: Users who rely on keyboard-only navigation cannot access significant portions of the search and filtering functionality, limiting their ability to refine search results and find relevant content.

Neurodiverse Users

- Issues: Heading hierarchy on the Home page starts at H4 rather than H1, disrupting the logical document outline. The Book page also lacks an H1. Navigation menus announce ambiguous states (links described as "collapsed" alongside buttons). Multiple dialogs open without notifying users, creating unpredictable page state changes.
- Impact: Users who rely on consistent page structure, predictable interaction patterns, and clear navigation landmarks may find the platform's heading hierarchy and widget behavior disorienting, increasing cognitive load during research tasks.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Page-Specific Findings and Impact Analysis

The following section documents accessibility opportunities identified by page. Findings are organized in ascending order by WCAG 2.2 success criteria number to support prioritized remediation planning.

Home Page

Opportunity Area	WCAG Success Criteria	Description	Example
Missing Alternative Text on Carousel Images	1.1.1 Non-text Content (Level A)	Book cover images in both featured carousels are read as file references (e.g., "B-9798700") rather than descriptive text. These numeric strings provide no meaningful information about the content being presented.	JAWS reads: "B-9798,700" for a book cover image. No title, author, or subject information is conveyed.
Heading Hierarchy Begins at H4	1.3.1 Info and Relationships (Level A)	The page heading structure starts at heading level 4 with no H1, H2, or H3 present. This prevents screen reader users from building an accurate mental model of the page via heading navigation.	JAWS heading navigation reveals: first heading announced as "Heading Level 4" with book titles. No H1 or H2 present on the page.
Carousel Widget Missing Accessible Name and Proper ARIA Structure	1.3.1 Info and Relationships (Level A) | 4.1.2 Name, Role, Value (Level A)	The carousel container uses role="listbox" but lacks an aria-label, making it impossible for screen reader users to identify the widget. Child elements do not conform to the required listbox child roles, and the listitem role on navigation items lacks a required parent list element.	Source: with no aria-label attribute.
Carousel Navigation Controls Keyboard Inaccessible	2.1.1 Keyboard (Level A)	The previous and next navigation buttons for the first carousel are not reachable by keyboard. The stop/pause control appears after the carousel content in the DOM, requiring significant navigation to reach. Carousel indicator buttons are labeled as "cell indicator 0" through "cell indicator 9" with no relationship to the slide content.	Keyboard testing confirmed previous/next buttons for carousel 1 are not in the tab order. Indicator buttons read as "cell indicator 0 button" with no descriptive label.
Navigation Submenu Redundant Link Elements and Missing ARIA States	4.1.2 Name, Role, Value (Level A)	Each navigation item (Subjects, Discover, Open Access, About, For Librarians) contains both a button and a separate link labeled "collapsed [item name]." The link element serves no navigation	Source: <li class="active" role="listitem"> missing required parent list role. JAWS announces:

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Opportunity Area	WCAG Success Criteria	Description	Example
		purpose and creates confusion. Buttons lack aria-expanded state changes when submenus open, and the spacebar key does not activate them.	"Subjects button. Link collapsed subjects."

Impact Summary: The combination of missing image alternative text, a broken heading hierarchy, and inaccessible carousel controls creates a significant barrier for screen reader users attempting to browse featured content on the home page. A user navigating via JAWS encounters book cover images read as numeric strings, cannot determine the page's organizational structure via headings, and may be unable to advance or stop auto-playing carousels using only a keyboard. The navigation issues compound this by making submenu exploration confusing and inconsistent. Resolving these opportunities will give screen reader users the same ability to browse featured titles that sighted users enjoy.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Home Page Screenshot

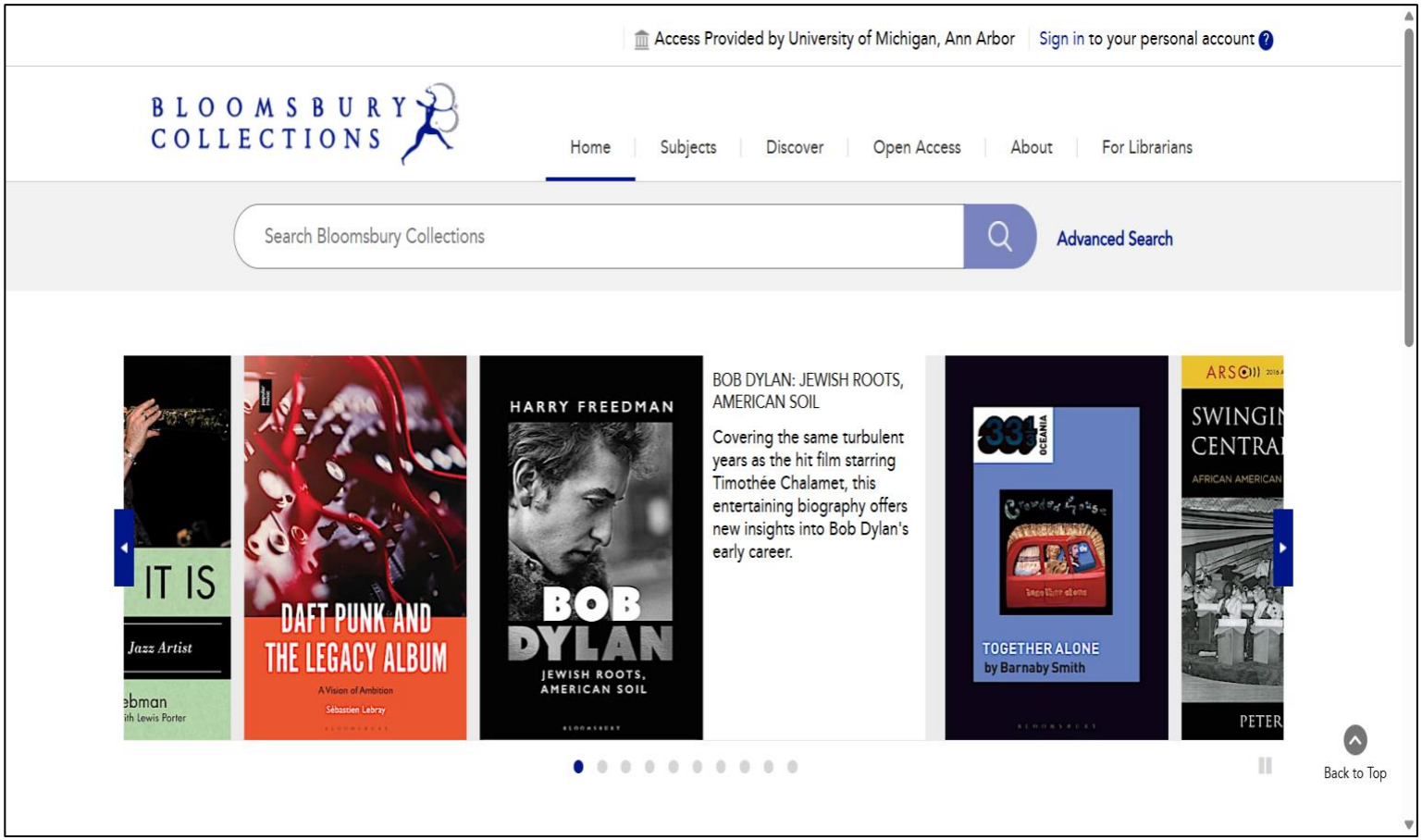

Figure 1. Bloomsbury Collections home page showing featured book carousel with auto-advance controls and primary navigation menu.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Search Results Page

Opportunity Area	WCAG Success Criteria	Description	Example
Book Cover Images Missing Alt Text	1.1.1 Non-text Content (Level A)	Multiple book cover thumbnail images on the Search Results page have missing or meaningless alternative text. Screen readers are unable to convey the title or subject of titles shown visually in search results.	Four instances of alt-text-missing flagged by automated scan on search results page. Informative images with no programmatic text alternative.
Filter Accordion Panels Use Invalid ARIA Attribute	1.3.1 Info and Relationships (Level A) | 4.1.2 Name, Role, Value (Level A)	The Refine Results filter panel uses aria-multiselectable="true" on a panel-group container that does not support this attribute for its role. This invalid attribute causes screen readers to misinterpret the filter panel's structure. A separate issue with aria-hidden="true" on informative result count content hides it from screen reader users.	Source: Three accordion instances flagged as critical.
Select Dropdowns Missing Accessible Names	4.1.2 Name, Role, Value (Level A)	The Sort By and Results Per Page select elements do not have programmatically associated labels. Screen reader users cannot determine the purpose of these dropdowns without additional context.	Source: select elements for sort and results-per-page controls without associated label elements or aria-label attributes.
Multiple Interactive Elements Not Keyboard Accessible	2.1.1 Keyboard (Level A)	Automated and manual testing identified numerous interactive controls on the Search Results page that cannot be operated by keyboard alone. These include filter controls, content interaction buttons, and result navigation elements.	Automated scan flagged 25+ instances of keyboard-inaccessible interactive elements across the page.
Focus Indicator Hidden on Filter Controls	2.4.7 Focus Visible (Level AA)	Several filter controls suppress the visible keyboard focus indicator using CSS, making it impossible for sighted keyboard users to track focus position within the Refine Results panel.	Seven instances of focus-indicator-missing flagged on filter panel elements. CSS suppresses the default browser focus ring without providing a visible alternative.

Impact Summary: The Search Results page is where users with disabilities most need a reliable, efficient experience, since finding relevant content is the primary task. The combination of keyboard-inaccessible filter controls, missing focus indicators, and invalid ARIA on the accordion filter panels means that keyboard users and screen reader users have limited ability to refine their searches. A screen reader user navigating the Refine Results panel encounters confusing ARIA structure and may not be able to interact with filters at all via

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

keyboard. Addressing these issues will bring the search and filtering experience in line with the accessibility of the core search bar, which performed well during testing.

Search Results Page Screenshot

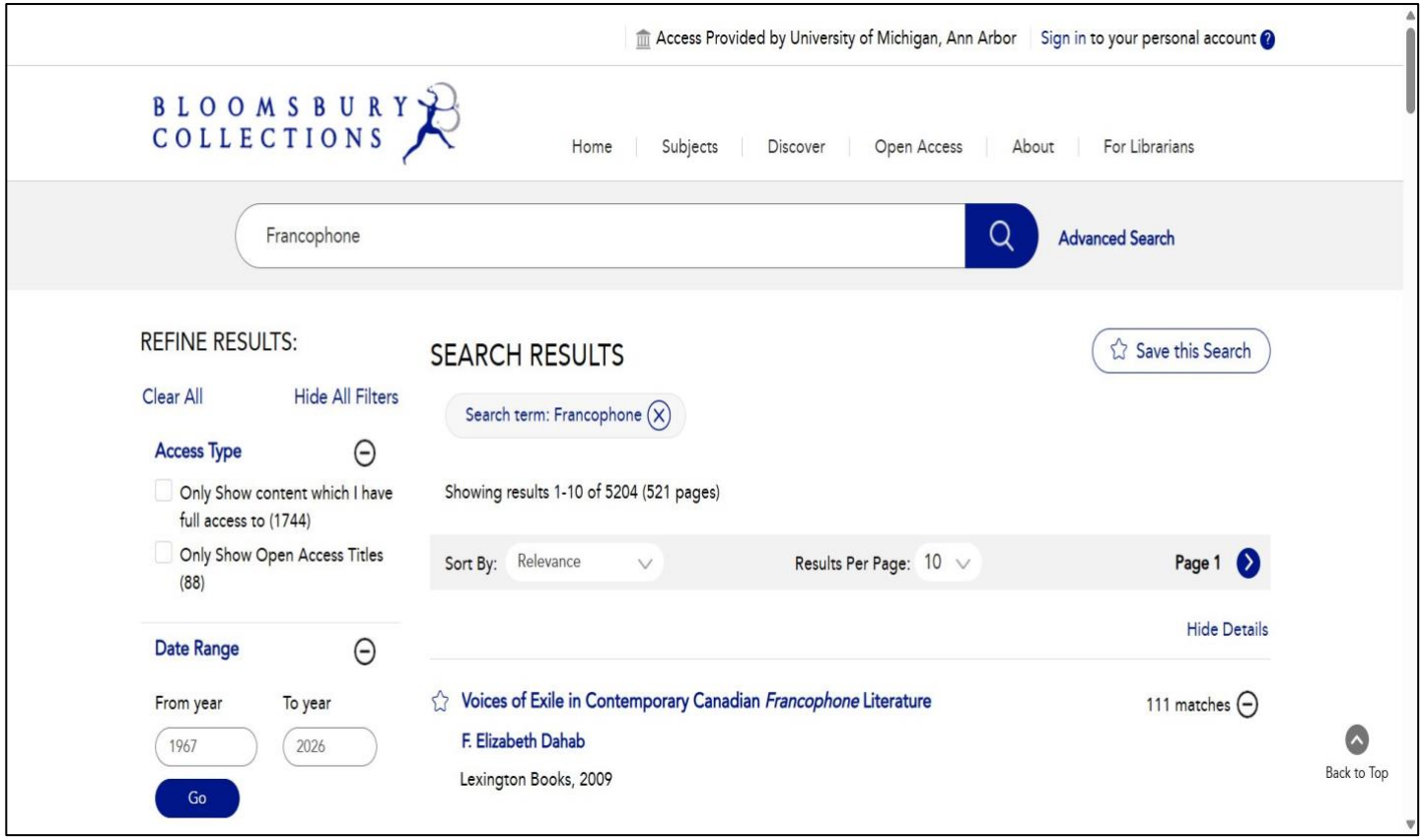

Figure 2. Search Results page showing a Francophone search query with the Refine Results filter panel on the left and search results on the right.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Advanced Search Page

Opportunity Area	WCAG Success Criteria	Description	Example
Form Group Labels Not Programmatically Associated	1.3.1 Info and Relationships (Level A)	The Year and Content Types section labels are implemented as span elements with tabindex rather than as proper HTML label or fieldset/legend elements. The label for the search dropdown ("Anywhere" select) is not programmatically tied to the control, requiring screen reader users to navigate separately to determine the field's purpose.	JAWS testing: auditor reports the label is read separately from the dropdown element. Source: used as a visual label with no programmatic association to form controls.
Non-interactive Elements Placed in Tab Order	2.4.3 Focus Order (Level A)	The H2 heading and Year span label have been given tabindex="0", placing non-interactive elements into the keyboard focus order. This creates unexpected focus stops that serve no purpose for keyboard users and may confuse screen reader users who expect focusable elements to be interactive.	Source: <h2 id="advanchsearch-title" tabindex="0"> and . Automated scan flagged 5 focus-order-semantics instances.
Help Dialog Does Not Announce as Modal and Allows Outside Reading	1.3.1 Info and Relationships (Level A) | 2.4.3 Focus Order (Level A)	The help tooltip/dialog on the Advanced Search page (activated by the question mark icon) reads as an H2 link rather than a button that opens a dialog. When activated, the dialog does not announce itself as a modal dialog, does not trap focus within the dialog, and allows the screen reader to continue reading content outside the open dialog.	JAWS testing: activating the help icon reads as an H2 link. Screen reader can navigate past the open dialog. Automated scan flagged reading-order-browse-outside-modal.
Add Field Button Loses Focus on Activation	4.1.2 Name, Role, Value (Level A)	When the Add Field button is activated to add a search row, keyboard focus is lost and not moved to the newly added field. The Add Field button also does not announce its disabled state when the maximum number of fields has been reached.	JAWS testing: activating Add Field causes focus loss. The button should become disabled with aria-disabled="true" when the limit is reached.
Checkbox Group Missing Visible Group Label	3.3.2 Labels or Instructions (Level A)	The Year range checkbox that enables/disables the year filter inputs does not have a visible group label explaining its purpose. The programmatic label does not descriptively convey the purpose of the checkbox control.	Source: <input id="yearcheckbox" type="checkbox"> with aria-label="year" which does not adequately describe the enable/disable function. Automated scan flagged label-group-not-present.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Impact Summary: The Advanced Search page offers powerful search capabilities that are particularly valuable for researchers, but several interaction barriers limit its usefulness for keyboard and screen reader users. When the page's own heading and section labels are placed into the tab order as non-interactive stops, keyboard navigation becomes unnecessarily cumbersome. The unannounced help dialog and focus management gaps mean users can get disoriented during what should be a straightforward search configuration workflow. Resolving the label associations, focus order, and dialog management issues will make Advanced Search as useful for screen reader users as it is for sighted users.

Advanced Search Page Screenshot

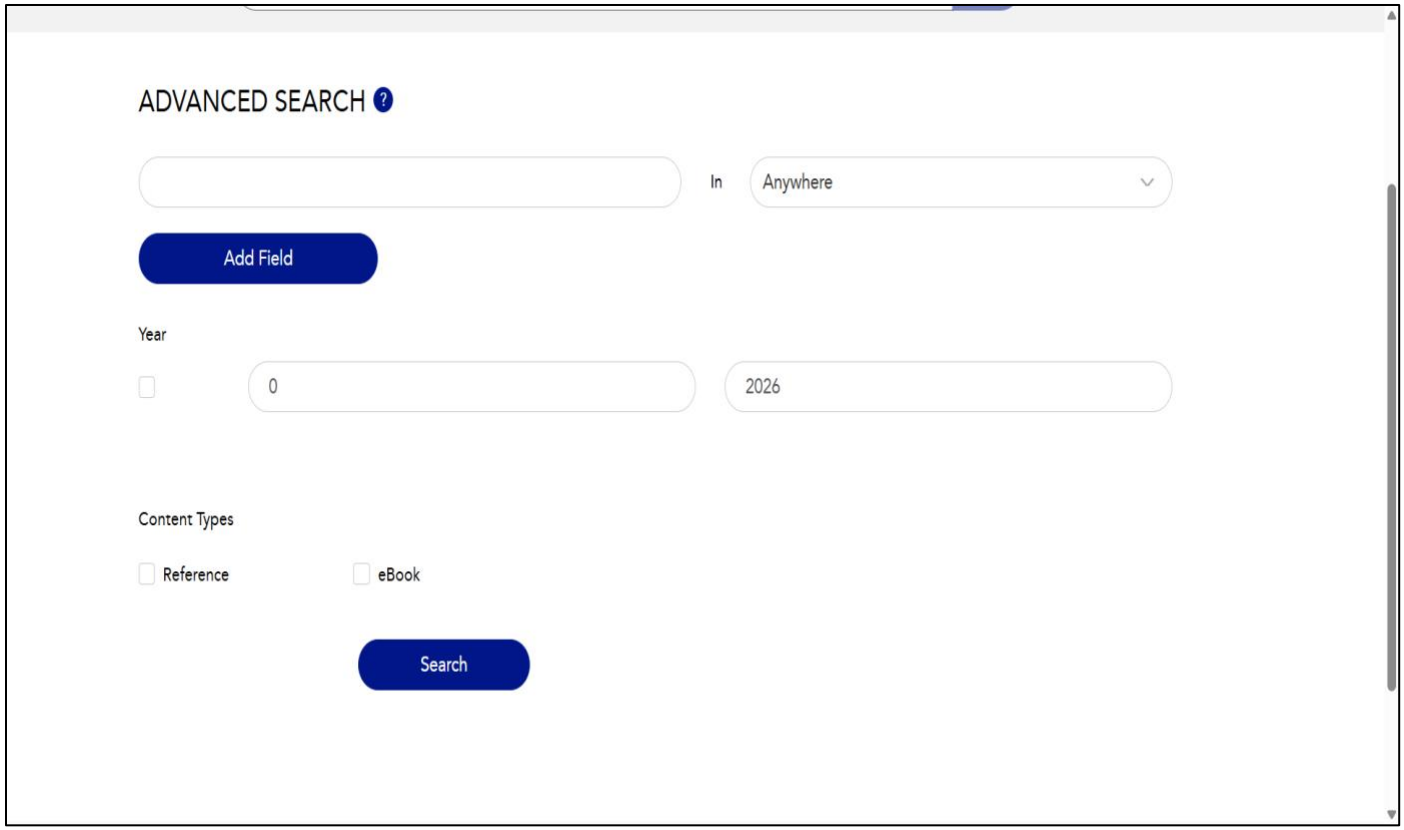

Figure 3. Advanced Search page showing the search input with an 'Anywhere' scope dropdown, Year range fields, Content Type checkboxes, and a Search button.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Book/Product Page

Opportunity Area	WCAG Success Criteria	Description	Example
Download Button Hidden from Assistive Technology While Remaining Focusable	1.3.1 Info and Relationships (Level A) | 4.1.2 Name, Role, Value (Level A)	The Download Full Book PDF button has aria-hidden="true" applied while remaining focusable and operable. This creates a contradictory state: keyboard users can tab to the button but screen readers will not announce it, making the download function effectively invisible to screen reader users.	Source: <a id="download" aria-hidden="true" role="button" href="..."> Download Full Book PDF. Keyboard focus reaches this element but screen readers skip it.
Table of Contents List Structure Invalid	1.3.1 Info and Relationships (Level A)	The Table of Contents accordion on the Book page uses a ul element with direct div and nested ul children, violating the HTML list structure requirements. This structural error causes screen readers to misreport the list and its items, making it difficult for users to gauge their location within the table of contents.	Source: <ul id="sub-accordion"> with direct div and ul children. JAWS testing: auditor notes inability to determine list boundaries clearly.
Pagination Input Missing Label	4.1.2 Name, Role, Value (Level A)	The "Go to page" input field in the pagination control has no programmatic label. Screen reader users encounter an unlabeled text field with no indication of its purpose.	Source: <input type="text" id="goto_pageno" class="form-control positive-integer number-class"> with no label element, aria-label, or aria-labelledby attribute.
Modal Dialogs Do Not Announce Role or Manage Focus	2.4.3 Focus Order (Level A) | 4.1.2 Name, Role, Value (Level A)	The Share, Print, and Citation overlay dialogs do not announce themselves as modal dialogs when activated. Focus is not moved into the dialog on open, and upon close, focus is returned to the top of the page rather than the triggering element. This requires keyboard users to re-navigate the entire page after each modal interaction.	JAWS testing: activating Share, Print, and Citation buttons does not trigger a modal announcement. After closing, auditor reports focus moves to top of page.
Tab Panel Announces All Tabs as Selected	4.1.2 Name, Role, Value (Level A)	The Citation dialog tab panel incorrectly communicates all tab options as selected simultaneously. The aria-selected state is not correctly managed, so screen reader users cannot determine which citation format tab is currently active.	JAWS testing: all citation format tabs are announced as selected. Source: tab elements missing correct aria-selected="false" on non-active tabs and aria-selected="true" on the active tab.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Impact Summary: The Book/Product page is the destination users reach after successfully searching and selecting a title, making it critical that all core functions are accessible. The download function is effectively invisible to screen reader users due to the aria-hidden conflict. The modal dialogs for citing, sharing, and printing all share the same focus management pattern that leaves users disoriented. The invalid list structure in the table of contents prevents efficient navigation of chapter listings. Fixing these issues will ensure that after a user locates a title, they can actually use the platform's core features: downloading, citing, and navigating the book's content.

Book/Product Page Screenshot

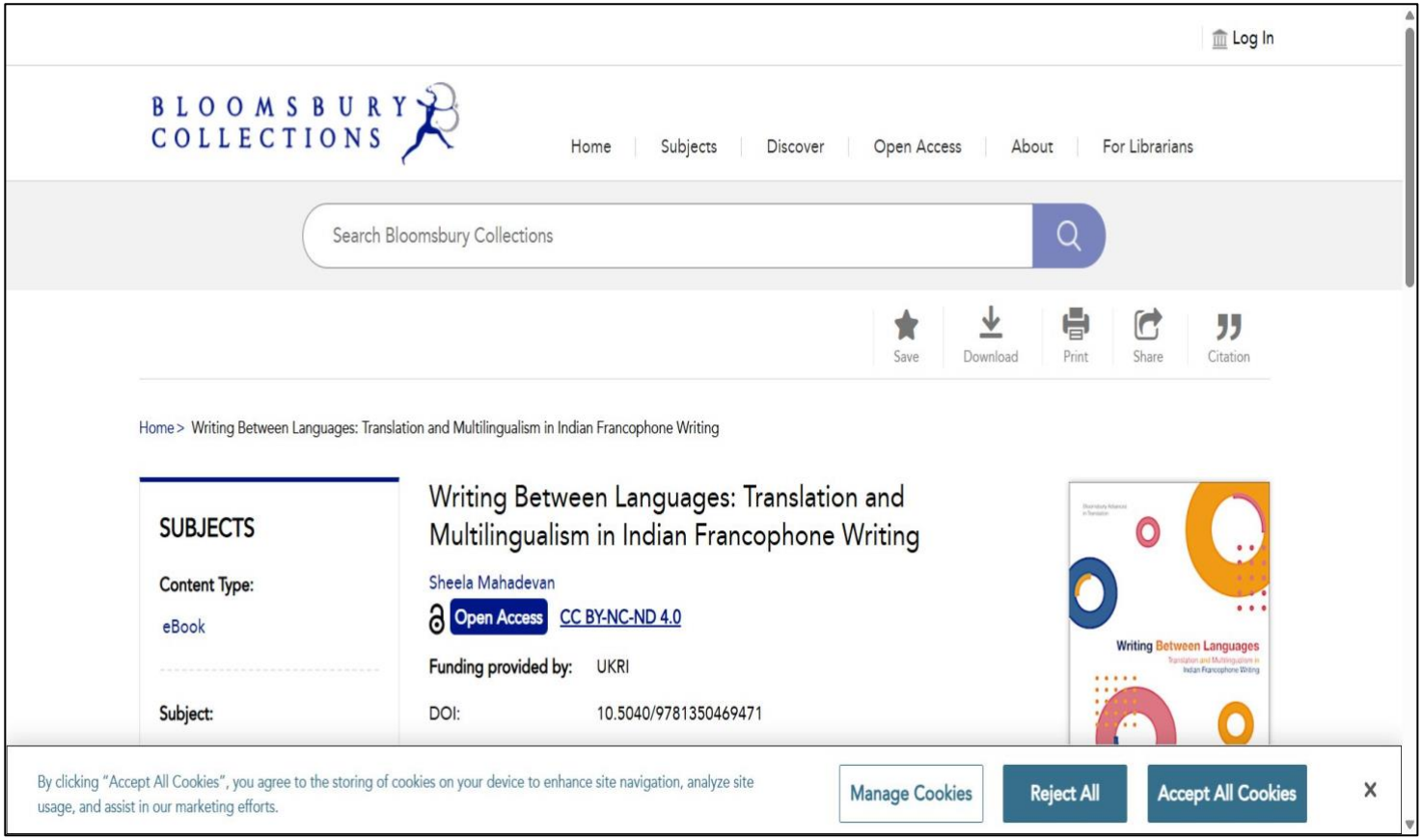

Figure 4. Book/Product page for 'Writing Between Languages' showing the title, author, Open Access badge, and action icons for Save, Download, Print, Share, and Citation.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Code Recommendations and Technical Guidance

The following code examples address the five most impactful accessibility opportunities identified across the evaluated pages. Each recommendation uses an HTML-first approach. Developers have flexibility to implement equivalent solutions using CSS and JavaScript alternatives, as long as the underlying accessibility goals are met.

Testing requirement: All code changes should be tested with JAWS and NVDA screen readers and keyboard-only navigation before deployment. These recommendations are guidance, not guaranteed fixes, and platform-specific implementation details may require adjustments.

1. Carousel: Add Accessible Name, Proper Role, and Keyboard Controls

Current implementation (problematic):

```
<!-- Carousel container has role="listbox" but no accessible name -->
<!-- Child items do not conform to listbox child role requirements -->
<div class="carousel-inner" role="listbox">
  <h1 tabindex="0">...</h1>  <!-- heading as listbox child is invalid -->
  <a tabindex="0">...</a>    <!-- link as direct child of listbox is invalid -->
</div>

<!-- Carousel position indicators have no meaningful labels -->
<button>cell indicator 0</button>
<button>cell indicator 1</button>
```

Recommended implementation:

```
<!-- Use role="region" with aria-label, or a proper ARIA carousel pattern -->
<section aria-label="Featured Books Carousel" aria-roledescription="carousel">
  <div aria-live="polite" aria-atomic="false">
    <!-- Each slide should be role="group" with aria-label -->
    <div role="group" aria-roledescription="slide" aria-label="1 of 10: Bob Dylan: Jewish Roots, American Soil">
      <img src="..." alt="Bob Dylan: Jewish Roots, American Soil by Harry Freedman - book cover" />
    </div>
  </div>
  <!-- Controls must be keyboard-accessible and descriptively labeled -->
  <button aria-label="Previous slide">&#8249;</button>
  <button aria-label="Next slide">&#8250;</button>
  <button aria-label="Pause carousel rotation" id="carousel-pause">&#10074;&#10074;</button>

  <!-- Indicators should describe their destination -->
  <button aria-label="Go to slide 1: Bob Dylan">&#9679;</button>
  <button aria-label="Go to slide 2: Daft Punk">&#9679;</button>
</section>
```

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

2. Navigation: Fix Button/Link Pattern and Add ARIA Expanded State

Current implementation (problematic):

```
<!-- Nav items have both a button AND a redundant link with no href purpose -->
<button class="nav-item">Subjects</button>
<a tabindex="0">link collapsed subjects</a>  <!-- This link is confusing and unnecessary -->
```

Recommended implementation:

```
<!-- Remove the redundant link, keep only the button with proper ARIA state -->
<nav aria-label="Main Navigation">
  <ul>
    <li>
      <a href="/home">Home</a>
    </li>
    <li>
      <!-- Button controls the submenu; aria-expanded reflects open/close state -->
      <button aria-expanded="false" aria-controls="subjects-menu" id="subjects-btn">
        Subjects
      </button>
      <ul id="subjects-menu" aria-labelledby="subjects-btn" hidden>
        <li><a href="/subjects/arts">Arts</a></li>
        <!-- additional submenu items -->
      </ul>
    </li>
  </ul>
</nav>

<!-- JavaScript: toggle aria-expanded and hidden on button click -->
<!-- Both spacebar and enter key should activate the button -->
```

3. Advanced Search: Associate Labels Programmatically

Current implementation (problematic):

```
<!-- span used as a visual label with no programmatic association to the input -->
<span class="label-txt" id="advansear-yr" tabindex="0" aria-labelledby="advansear-yr">Year</span>
<!-- The select dropdown has no label -->
<select>
  <option>Anywhere</option>
</select>
```

Recommended implementation:

```
<!-- Use fieldset/legend for grouped controls; use label for individual inputs -->
<fieldset>
  <legend>Year Range</legend>
  <label for="year-from">From year</label>
  <input type="number" id="year-from" name="year-from" value="0" min="0" max="2026" />
```

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

```
  <label for="year-to">To year</label>
  <input type="number" id="year-to" name="year-to" value="2026" min="0" max="2026" />
</fieldset>

<!-- For the search scope dropdown -->
<label for="search-scope">Search in</label>
<select id="search-scope" name="scope">
  <option value="anywhere">Anywhere</option>
  <option value="title">Title</option>
  <option value="author">Author</option>
</select>

<!-- Remove tabindex from non-interactive heading -->
<h2 id="advanchsearch-title">ADVANCED SEARCH
  <button aria-label="Advanced Search help" aria-haspopup="dialog"
          data-toggle="modal" data-target="#help-content">?</button>
</h2>
```

4. Modal Dialogs: Add Role, Aria-Modal, and Focus Management

Current implementation (problematic):

```
<!-- Modal dialog does not announce as dialog, doesn't trap focus -->
<div id="share-modal" class="modal fade">
  <!-- Content here, but no dialog role, no aria-modal, no labelledby -->
  <button class="close">x</button>
</div>
```

Recommended implementation:

```
<!-- Add role="dialog", aria-modal, and aria-labelledby -->
<div id="share-modal" role="dialog" aria-modal="true"
     aria-labelledby="share-modal-title" aria-describedby="share-modal-desc">
  <h2 id="share-modal-title">Share This Title</h2>
  <p id="share-modal-desc">Use the options below to share this title.</p>
  <!-- modal content -->
  <button id="share-modal-close">Close</button>
</div>

<!-- JavaScript requirements:
  1. On modal open: move focus to the first interactive element inside the dialog
  2. Trap Tab/Shift+Tab within the dialog while it is open
  3. Close on Escape key press
  4. On modal close: return focus to the triggering element (e.g., the Share button)
-->
```

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

5. Download Button: Remove aria-hidden Conflict

Current implementation (problematic):

```
<!-- aria-hidden="true" hides the button from screen readers while it remains focusable -->
<a id="download" aria-hidden="true" role="button"
   aria-label="Download Full Book PDF" href="/app/downloadpdf?..."
   target="_blank" download="...">
  <!-- icon here -->
</a>
```

Recommended implementation:

```
<!-- Remove aria-hidden so the button is both visible and announced by screen readers -->
<a id="download" role="button"
   aria-label="Download Full Book PDF (opens in new tab)"
   href="/app/downloadpdf?..." target="_blank" download="..."
   rel="noopener noreferrer">
  <!-- icon here -->
</a>

<!-- If the button should be conditionally hidden (e.g., not available for this title),
     use: aria-disabled="true" and remove the href, or remove from DOM entirely.
     Do not use aria-hidden on focusable elements. -->
```

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Final Thoughts and Recommendations

Bloomsbury Collections has a well-organized platform with clear information architecture and a strong foundation to build on. The presence of landmark regions, a functioning skip link, and generally good color contrast demonstrates that accessibility has been considered in the platform's design. The issues identified in this evaluation are largely systemic patterns rather than isolated edge cases, which is actually good news: fixing a pattern in one place typically resolves similar issues across the entire platform.

Recommended Fixes by Priority

Immediate Priority

- Remove aria-hidden="true" from the Download Full Book PDF button. This is a single attribute change that immediately restores screen reader access to a core feature.
- Add aria-label attributes to the sort and results-per-page select elements on the Search Results page. Minimal code change with immediate impact on search usability.
- Add aria-label to the pagination input on the Book page.
- Remove tabindex from non-interactive elements (headings, spans) on the Advanced Search page.

High Priority

- Implement proper modal dialog patterns (role="dialog", aria-modal, focus trap, Escape key, and focus return) across all dialogs on the Book page and Advanced Search page. One reusable modal component can resolve all dialog issues site-wide.
- Fix the navigation dropdown pattern: remove redundant link elements, keep buttons with proper aria-expanded state, and ensure both spacebar and enter activate them.
- Add meaningful alternative text to book cover images in carousels and search results. Even a simple pattern of "[Title] by [Author] - book cover" would be a significant improvement.
- Address invalid ARIA attribute usage on filter accordion panels on the Search Results page (aria-multiselectable on non-supporting elements).

Important Priority

- Rebuild the carousel component using the ARIA Authoring Practices carousel pattern, or a well-tested accessible carousel library. This addresses the compound issues of missing accessible name, role structure, keyboard navigation, and indicator labels.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

- Fix the heading hierarchy to start at H1 on the Home page and Book page. This enables screen reader users to navigate and scan page content efficiently.
- Correct the table of contents list structure on the Book page to ensure screen readers accurately report list boundaries.
- Programmatically associate form labels on the Advanced Search page using label elements or fieldset/legend for grouped controls.
- Ensure keyboard-inaccessible interactive elements on the Search Results page are reachable and operable via keyboard alone.

Advantages of Addressing These Opportunities

For academic libraries and their institutions, accessible digital resources are not optional. Section 508, the ADA, and increasingly state-level legislation require that digital platforms provided to students, faculty, and staff meet WCAG 2.2 AA standards. Libraries evaluating platforms for procurement increasingly use accessibility as a key criterion, and platforms with demonstrable accessibility commitments are better positioned in that evaluation process.

Beyond compliance, the estimated 26% of adults in the US with some form of disability represents a substantial portion of the academic user base. Accessible design also benefits users without disabilities: clear heading structure, keyboard accessibility, and well-labeled controls improve usability for all users, particularly those accessing the platform in challenging conditions (mobile, slow connections, or noisy environments).

The fixes recommended here are largely code-level changes that do not require visual redesign. Most can be implemented incrementally, with the highest-impact items achievable in a single development sprint. A systematic approach to the ARIA patterns and modal components would resolve a significant portion of the issues identified across all four evaluated pages.

Complimentary Consultation Included

As part of this evaluation, Accessiblü's partnership with the LAA includes one hour of complimentary consulting with the team that conducted this evaluation. This session can be used to discuss implementation priorities, review technical approaches, or address questions about specific recommendations. To schedule this consultation, contact Jeff Rodgers directly at jeff@accessiblu.com.

882 Pompton Ave. STE A-2
Cedar Grove NJ, 07009

Disclaimer

Accessiblü prepared this report as a high-level accessibility evaluation of the Bloomsbury Collections platform. The evaluation utilized industry-standard testing methodologies, including screen reader testing (JAWS 2025), keyboard-only navigation, manual inspection, and automated scanning via axe DevTools for select WCAG 2.2 AA success criteria.

This report does not represent a comprehensive WCAG compliance audit and should not be considered a certification of accessibility compliance. While significant accessibility opportunities and usability barriers have been identified, this evaluation was limited in scope and may not encompass all accessibility issues present on the platform.

No Legal Liability

Accessiblü offers this report for informational purposes only. It assumes no legal responsibility for accessibility barriers or compliance requirements resulting from the use of this report. Organizations seeking formal certification should conduct a comprehensive audit and engage in user testing with individuals with disabilities.

Limitations of Testing

This evaluation was conducted on February 10-11, 2026. Platform updates may have occurred after testing was completed. While automated tools and expert manual review were utilized, real-world users with disabilities are the true measure of accessibility. Automated tools typically identify a subset of accessibility issues; this report should be considered a starting point rather than an exhaustive inventory.

Testing was conducted via a university library proxy (University of Michigan, Ann Arbor). Some platform behaviors may differ for direct access users, authenticated personal account users, or users accessing the platform through different institutional configurations.