

VPAT Accessibility Conformance Report

(Based on ITI VPAT®)

Name of Product	Reaxys
Date Last Updated	July 14, 2026
Completed by	Christopher Merrington (Elsevier Digital Accessibility Team) Nicola Richardson (Elsevier Digital Accessibility Team)
Applicable Standards/Guidelines	This document rates Reaxys according to the W3C WCAG 2.1 A and AA requirements.
Contact for More Information	Elsevier Digital Accessibility Team accessibility@elsevier.com
Testing Tools and Methods	• Hands-on keyboard operation • DevTools/Code inspection • Safari (26.1) on macOS 15.7.2 and Chrome (143.0.7499.170) on Windows 11 23H2 • NVDA screen reader (2025.3.2) • VoiceOver screen reader (15.7.2) • Color Contrast Analyzer • W3C Web Accessibility Initiative (WAI) Pages • Elsevier Accessibility Checklist
Document Sections	This review document includes all WCAG 2.1 A and AA checkpoints, organized into 7 logical sections: • Visuals • Keyboard • Headings and Structure • Labeling • Multimedia • Usability • Mobile User Experience
Pages Covered	LogIn (sign in, register and Institution login), Header, Footer, Quick Search, Quick Search Results, Query Builder, Document Results, Targets Results, My retrosynthesis projects, History, Structure Editor and Alerts
Terms	• Supports: The functionality of the product has at least one method that meets the criteria without known defects or meets with equivalent facilitation. • Partially supports: Some functionality of the product does not meet the criteria. • Does not support: Majority of functionality of the product does not meet the criteria. • Supports (N/A): According to W3C on conformance, "If there is no content to which a success criterion applies, the success criterion is satisfied."
Notes/Terminology	• "AT" stands for Assistive Technology such as screen readers, voice input, etc.

Conformance Summary

WCAG 2.1 Success Criterion	Level	Evaluation
1.1.1: Non-text Content	A	Supports
1.2.1: Audio-only and Video-only (Prerecorded)	A	Supports (N/A)
1.2.2: Captions (Prerecorded)	A	Supports (N/A)
1.2.3: Audio Description or Full Text Alternative	A	Supports (N/A)
1.2.4: Captions (Live)	AA	Supports (N/A)
1.2.5: Audio Description	AA	Supports (N/A)
1.3.1: Info and Relationships	A	Partially supports
1.3.2: Meaningful Sequence	A	Supports
1.3.3: Sensory Characteristics	A	Supports (N/A)
1.3.4: Orientation (2.1)	AA	Supports
1.3.5: Identify Input Purpose (2.1)	AA	Supports
1.4.1: Use of Color	A	Supports
1.4.2: Audio Control	A	Supports (N/A)
1.4.3: Contrast (Minimum)	AA	Supports
1.4.4: Resize text	AA	Partially supports
1.4.5: Images of Text	AA	Supports
1.4.10: Reflow (2.1)	AA	Does not support
1.4.11: Non-Text Contrast (2.1)	AA	Partially supports
1.4.12: Text Spacing (2.1)	AA	Supports
1.4.13: Content on Hover or Focus (2.1)	AA	Partially supports
2.1.1: Keyboard	A	Partially supports
2.1.2: No Keyboard Trap	A	Supports
2.1.4: Character Key Shortcuts (2.1)	A	Supports (N/A)
2.2.1: Timing Adjustable	A	Partially supports
2.2.2: Pause, Stop, Hide	A	Supports
2.3.1: Three Flashes or Below Threshold	A	Supports (N/A)
2.4.1: Bypass Blocks	A	Supports
2.4.2: Page Titled	A	Supports
2.4.3: Focus Order	A	Partially supports
2.4.4: Link Purpose (In Context)	A	Supports
2.4.5: Multiple Ways	AA	Supports
2.4.6: Headings and Labels	AA	Partially supports
2.4.7: Focus Visible	AA	Partially supports
2.5.1: Pointer Gestures (2.1)	A	Supports (N/A)
2.5.2: Pointer Cancellation (2.1)	A	Supports
2.5.3: Label in Name (2.1)	A	Partially supports
2.5.4: Motion Actuation (2.1)	A	Supports (N/A)
3.1.1: Language of Page	A	Supports
3.1.2: Language of Parts	AA	Does not support

WCAG 2.1 Success Criterion	Level	Evaluation
3.2.1: On Focus	A	Partially supports
3.2.2: On Input	A	Supports
3.2.3: Consistent Navigation	AA	Supports
3.2.4: Consistent Identification	AA	Supports
3.3.1: Error Identification	A	Partially supports
3.3.2: Labels or Instructions	A	Partially supports
3.3.3: Error Suggestion	AA	Supports
3.3.4: Error Prevention (Legal, Financial, Data)	AA	Supports (N/A)
4.1.1: Parsing	A	Supports
4.1.2: Name, Role, Value	A	Partially supports
4.1.3: Status Messages (2.1)	AA	Partially supports

WCAG 2.1 A and AA Success Criteria

Visuals

WCAG 2.1 Checkpoint	Conformance Level	Remarks
1.1.1: Non-Text Content (A) Provide text alternatives for non-text content (e.g. images)	Supports	Images and icons have appropriate text equivalents. Chemical structures, where known, have relevant text alternatives provided. There may be instances where information about the structure is not known (e.g. for user-drawn structures), the text alternative in this instance indicates the origin of the image.
1.3.3: Sensory Characteristics (A) Do not rely on sensory characteristics of components such as shape, size, visual location, orientation, or sound	Supports (N/A)	There are no instructions or areas of content which rely solely on sensory characteristics.
1.4.1: Use of Color (A) Color is not used as the only visual means of conveying info	Supports	When color is used as a means of conveying information, another visual method is also used to convey the information without color.
1.4.3: Color Contrast (Minimum) (AA) Text has enough contrast with the background (4.5:1 for small text and 3:1 for large text)	Supports	Text has sufficient contrast with its corresponding background.
1.4.4: Resize Text (AA) Text can be enlarged up to 200% without loss of functionality.	Partially supports	Text may be enlarged to 200% while preserving functionality of content in most instances. Exceptions: My retrosynthesis projects: When text size is increased to 200%, buttons to edit a drawn structure and actions for the structure overlap and are indistinguishable.
1.4.5: Images of Text (AA) Text is used rather than images of text, except where the presentation of text is essential, such as logos	Supports	No images of text are used other than for logos or essential presentation in most areas.
1.4.10: Reflow (AA) Content can be presented without loss of information or functionality, and	Does not support	Many pages cannot be reflowed without losing content, functionality or requiring scrolling in two dimensions to access content. Content is obscured or lost in the following areas: Footer, Quick search results, Query builder, Structure editor and most dialogs.

without requiring scrolling in two dimensions for: - Vertical scrolling content at a width equivalent to 320 CSS pixels; - Horizontal scrolling content at a height equivalent to 256 CSS pixels..		Remarks: - Content does appropriately reflow for Login (Sign in, Register and Institution login) and Header Quick search: the page largely reflows with some clipping of instructional text Alerts: main page content reflows; dialog content does not Documents: main page content reflows with some exceptions in filters and AI summary tables History: main page content reflows but the 'rename' control is obscured by a button
1.4.11: Non-Text Contrast (AA) User interface components and graphical objects have a contrast ratio of at least 3:1 against adjacent color(s).	Partially supports	Almost all non-text UI components and graphical objects have at least a 3:1 contrast ratio against surrounding colors. Exceptions: Structure editor: On toolbar items focus indicators may be difficult to perceive due to color combination used
1.4.12: Text Spacing (AA) In content implemented using markup languages that support the following text style properties, no loss of content or functionality occurs by setting all the following and by changing no other style property: - Line height (line spacing) to at least 1.5 times the font size; - Spacing following paragraphs to at least 2 times the font size; - Letter spacing (tracking) to at least 0.12 times the font size; - Word spacing to at least 0.16 times the font size.	Supports	Users may adjust the text spacing of content on pages to the minimum baseline properties without causing loss of content or functionality.
1.4.13: Content on Hover or Focus (AA) Where receiving and then removing pointer hover or keyboard	Partially supports	Tooltip content appears throughout the product. For the most part this content is hoverable and persistent. Documents, Targets and My retrosynthesis projects: Some tooltips cannot be dismissed using the keyboard

focus triggers additional content to become visible and then hidden, the following are true: • Dismissible • Hoverable • Persistent		
2.3.1: Three Flashes or Below Threshold (A) No more than three flashes in a 1-second period, or the flashes are below the defined thresholds	Supports (N/A)	No flashing content exists.

Keyboard

WCAG 2.1 Checkpoint	Conformance Level	Remarks
1.3.2: Meaningful Sequence (A) The correct reading sequence can be programmatically determined	Supports	The correct reading sequence is logical and programmatically determinable, with the DOM order according with the visual order.
2.1.1: Keyboard (A) All functionality is available from a keyboard, except for tasks such as drawing	Partially supports	Most standard web page content and functionality is keyboard operable across pages. Exceptions: • Query builder: Sections can be dragged using the mouse to add them to the form and combine sections. While a keyboard alternative is provided for basic reorganization actions, not all behavior available to mouse users is available to keyboard users.
2.1.2: No Keyboard Trap (A) The user can use the keyboard to move through page elements and is not trapped on a particular element	Supports	No pages have a keyboard trap.
2.1.4: Character Key Shortcuts (A) If a keyboard shortcut is implemented in content using only letter (including upper- and lower-case letters), punctuation, number, or symbol characters, then at	Supports (N/A)	The site does not use any character key shortcuts.

least one of the following is true: • Turn off • Remap • Active only on focus		
2.4.3: Focus Order (A) Users can tab through the elements of a page in a logical order	Partially supports	Tab order is largely logical across the site and preserves the meaning and operability of content in most instances. There are some instances where non-interactive elements have been erroneously added to the page focus order to enable keyboard access to tooltips. Exceptions: • Footer: When navigating between steps in the 'Remote access' dialog, keyboard focus is not managed correctly and is set to the top of the document rather than the subsequent dialog elements.
2.4.7: Focus Visible (AA) The page element with the current keyboard focus has a visible focus indicator	Partially supports	Almost all elements across the site have a decent visible indication of focus – the focus indicator is typically a solid orange outline. Exceptions: • Documents and Targets: When interacting with pagination buttons, the 'Go to page' input, or the 'Results per page' control, focus is moved to the results container without providing a visible focus indicator. • Structure editor: Buttons to clear data from inputs in the side panel do not have visible focus indicators.
3.2.1: On Focus (A) When a UI component receives focus, this does not trigger unexpected actions.	Partially supports	Focusable elements do not generally cause unexpected actions/changes of context when receiving focus. Exceptions: • Query builder: Keyboard drag functionality is triggered on focus of the drag button, rather than activation.

Headings and Structure

WCAG 2.1 Checkpoint	Conformance Level	Remarks
1.3.1: Information and Relationships (A) Info, structure, and relationships can be programmatically determined	Partially supports	Most content is distinguishable via semantic structure and relationships. A logical heading order reflecting page organization and content is programmatically determinable on most pages. List markup is used appropriately in many instances. Most input elements have programmatically determinable labels. HTML sectioning elements/landmark roles demarcate content regions. Exceptions: • My retrosynthesis projects, Alerts and Quick search results, Documents, Header and Targets: Text that visually looks and acts like heading text (text that defines a section of the page) is not marked up as such or does not accurately represent the page structure.

- **Create alert dialog:** Interactive elements that are visually related are not related programmatically and not announced in a way (via the screen reader) that maintains visual relationships.
- **Alerts, History, Documents, Structure editor and Quick search results:** Some elements on the page have the same accessible name with no method in place to allow screen reader users to quickly/easily differentiate between them.
- **Create alert dialog, My retrosynthesis projects, Header, Targets, Documents and Quick search results:** Infotip or popover content boundaries are not clear to screen reader users.
- **Create alert dialog – success, Quick search results and Documents:** Key–value pairs are presented visually, but there is no programmatic structure to convey these relationships.
- **My retrosynthesis projects, Documents and Targets:** Navigation landmark used inappropriately to contain pagination buttons.
- **Quick search results:** A search landmark is used to define sections of pages that primarily display search results rather than search functionality.
- **Query builder, Targets and Documents:** Additional information associated with checkbox elements, such as numeric values, is not announced when screen reader moves focus to the checkbox.
- **Quick search results:** Druglikeness table header cells span two columns but only describes the first column.
- **Header:** Number is not programmatically associated with the 'Resource center' button.
- **Header:** Nested navigation landmarks do not convey clear relationships.
- **Header:** The Reaxys wordmark is identified as a level one heading.
- **Quick search results:** The relationship between the text 'Structure:' and 'as drawn' is visually clear, but there is no programmatic association to ensure screen reader users understand they are related.
- **Quick search results:** Several references and corresponding links are present in a single cell. There is no code in place to

		allow screen reader users to understand which link belongs to which reference. • Query builder: In the query form, some programmatic structure is in place to convey groupings and hierarchy. However, this is likely insufficient to ensure screen reader users can fully understand the relationships between sections and content areas. • Query builder: In dialog, a nested set of items is presented visually as a list, however it is not announced as a list or as nested content by screen readers due to the markup in place. • Query builder: In dialog, heading structure does not convey the relationship between sections. • Query builder: Text such as 'Alert:' and Information: relates to visible message text but no code is in place to programmatically link the two text items. • Query builder: Tab panels associated with tab buttons that are not currently active can still be discovered by screen readers • Query builder, Documents, Targets, Alerts and My retrosynthesis projects: There is no <main> landmark present on the page. • Query builder: <nav> is used to contain elements that don't act like navigation elements (elements used to import/save and add form sections). • Targets: The relationship between certain text and associated links is visually clear, but there is no programmatic structure to convey these relationships to screen reader users. • Structure editor: A search landmark is used for a group of elements that do not collectively represent a complete search function. • Structure editor: Elements within the side panel are grouped using <section> elements that do not have accessible names. Unnamed sections are not exposed as meaningful landmarks to screen reader users. • Structure editor: Groups of related content, such as an image, text, and a button, are visually presented as distinct blocks. However, there is no programmatic structure to ensure screen reader users understand the relationship between these elements.
2.4.1: Bypass Blocks (A)	Supports	Each page features a "Skip to Main Content" skip navigation link.

Users can bypass repeated blocks of content.		
2.4.6: Headings and Labels (AA) Headings and labels are clear and consistent.	Partially supports	Headings and labels used are typically clear and descriptive. For example, most pages feature visually distinct and programmatically determinable main and secondary headings to help distinguish content. Exceptions: • Query builder, Documents, Targets, History, Structure editor, Header: Some dialogs have the accessible name 'Dialog overlapping the application content'. This label does not describe the purpose or content of the dialog. • Query builder and Structure editor: The accessible name of the control used to open and close a list of options does not describe the purpose of the control. Instead, it is labelled only with the currently selected value. • Quick search results and Targets: Some 'Close' buttons do not clearly communicate what they will close. Screen reader users who cannot see the screen may not understand the purpose of these controls, especially where multiple expandable sections and controls are present. • Quick search results and Documents: Some buttons have aria-label values that include code-like text such as <code><hi>Paracetamol</hi></code> when highlighted. Screen readers announce this content literally, which does not provide meaningful information to users. • Query builder: Buttons used to navigate between pages have accessible names that do not accurately describe their purpose. • My retrosynthesis projects: It is visually clear which elements belong to the 'Predicted' group and which belong to the 'Published' group. However, there is no programmatic structure in place to convey these groupings to screen reader users. • My retrosynthesis projects: The first column in the table includes a header cell that contains no content. • Structure editor: The main heading on the page (the <code><h1></code>) describes the radio buttons that follow rather than the purpose of the page as a whole.
3.1.1: Language of Page (A) The language of the page is specified	Supports	The default page language is typically and appropriately defined as <code>lang="en"</code> .

3.1.2: Language of Parts (AA) Specify the language of text passages that are in a different language than the default language of the page.	Does not support	For content in a language other than English, the language of the part is not specified.
4.1.1: Parsing (A) Use valid, error-free HTML	Supports	HTML and CSS typically pass concerning these 4 specific criteria: (i) elements have complete start and end tags, (ii) elements are nested according to their specifications (iii) elements do not contain duplicate attributes (iv) any IDs are unique, except where the specifications allow these features. Note: There may be other general HTML validation errors outside the scope of this criterion. WCAG 2.1 Errata notes: "This Success Criterion should be considered as always satisfied for any content using HTML or XML."

Labeling

WCAG 2.1 Checkpoint	Conformance Level	Remarks
1.3.5: Identify Input Purpose (AA) The purpose of each input field collecting information about the user can be programmatically determined when: - The input field serves a purpose identified in the Input Purposes for User Interface Components section; and - The content is implemented using technologies with support for identifying the expected meaning for form input data.	Supports	The only pages featuring applicable form elements that collect such information about the user are Login screens (sign in, register and Institution login). Autocomplete attributes are implemented to facilitate auto-fill on these screens.
2.4.2: Page Titled (A) The page has a title describing its topic or purpose	Supports	A descriptive page title that identifies content/purpose is not present for most pages: "Elsevier" is defined generically as the page title across many pages
2.4.4: Link Purpose (In Context) (A)	Supports	Links have descriptive and unique names or identifiable purpose may be deduced from the link text or surrounding context.

The purpose of each link can be determined from the link text or surrounding context.		
2.5.3: Label in Name (A) For user interface components with labels that include text or images of text, the name contains the text that is presented visually.	Partially supports	Most user interface components that have visible text contain that text consistently within the accessible name. Exceptions: • Documents, Targets and Quick search results: Some combobox elements have an accessible name that does not match the visible label provided. • Documents and Targets: The 'Results per page' dropdown has an accessible name that does not match the visible label. • Header: The button used to open profile popover has a visible label of 'CM' and a tooltip label of 'Reaxys access', however its accessible name is 'Profile'. • Quick search results: The button used to show hidden columns has a visible label that begins with '4...', but its accessible name does not include the number 4. • Alerts: The dropdown button has a visible label of 'Results from', but its accessible name is set to only the currently selected value.
3.2.4: Consistent Identification (AA) UI components used across the web site are identified consistently on every page.	Supports	Components are typically consistent across the site, and identified consistently where they perform the same function across pages.
3.3.1: Error Identification (A) Input errors are clearly marked and described to the user.	Partially supports	In most instances, errors are identified and presented well visually. For many inputs, errors are typically validated before form submission. Error messages that offer specific feedback are presented adjacently and visually distinguished via different text color (red). Error messages are announced to AT in some instances. Several other error states are not programmatically determinable. Exceptions: • My retrosynthesis projects: Errors are not always communicated in text, instead the user's change is not saved, and the project name is reverted to the original without indication something is wrong or suggestions to avoid the error.
3.3.2: Labels or Instructions (A) Items requiring user input are clearly labeled or have clear instructions.	Partially supports	Labels or instructions are provided for most form elements, most of which are programmatically associated with their inputs. Exceptions:

		• Create alert dialog, History and Login (Sign in and Register): Some fields that must be completed to submit or complete an action use an asterisk to indicate required status, but do not provide explicit explanation. • Structure editor: Placeholder text is used in place of a visible label for some inputs. Placeholder text typically disappears when users begin typing, which can make it harder for users to remember the purpose of the field. • Login (Institution login): Password requirements are not provided before validation.
3.3.3: Error Suggestion (AA) When the user makes an input error, give suggestions for valid input.	Supports	The nature of content would largely not give rise to opportunities for error suggestions, but relevant helpful suggestions are occasionally provided in text.
4.1.2: Name, Role, Value (A) For all UI components, the name, value, and role can be programmatically determined.	Partially supports	Some UI components communicate their state programmatically, and many have accessible names that are appropriately defined. Several ARIA attributes and roles are not present where appropriate. Exceptions: • Alerts, Query builder, Structure editor, History, My retrosynthesis projects, Quick search results, Documents, Targets and Footer: Many dialogs across the product do not have accessible names. • Create alert dialog: Some dropdown elements within the alert dialog do not have an accessible name. • Create alert dialog: Buttons used to reveal infotip content do not provide programmatic state information. • Documents, Targets: Some elements use the aria-current attribute with a value of 'page', even when they do not represent the current page. • My retrosynthesis projects, Targets, Quick search results and Documents: In popover content related to drawn structures, the roles menu and menuitem are applied to elements that do not function as application-style menus. • Quick search results: Section headers can be activated to open and close the content below them, but they are not consistently marked up as interactive controls. • Quick search results and Documents: Buttons used to show or hide sections have the same accessible name regardless of whether the section is currently open or closed. The name includes 'close' even when the section is already closed. There

		is no other code in place to ensure screen reader users understand the current state. • Sorting buttons: When some sort buttons are activated, there may not be feedback to confirm that sorting has occurred. • Targets and Documents: The 'Filter by value' button toggles between filtering by entering a value directly in an input and filtering by selecting checkboxes. However, there is no programmatic indication of the current state. • Quick search results, Alerts, Documents and Targets: Comboboxes are not marked up or announced correctly via the screen reader. • Header and My retrosynthesis projects: Buttons used to open and close popover content do not always expose their current state programmatically. • Header: Results is sometimes visually presented as disabled; however this state is not programmatically conveyed to screen reader users. • Header: Interactive controls which perform actions or navigate do not always have the most appropriate role, sometimes using a button role for navigation and vice versa. • Query builder: In dialog, buttons used to open and close sections do not expose their state programmatically and buttons used to clear selected items do not have clear accessible names. • Alerts: The 'Results from' dropdown is marked up as a button but does not expose any state information. • My retrosynthesis projects: When the last project is removed via a 'delete' button, it triggers unexpected navigation to /draw and a complete change of UI without warning.
4.1.3: Status Messages (AA) In content implemented using markup languages, status messages can be programmatically determined through role or properties such that they can be presented to the user by assistive	Partially supports	Status messages, while uncommonly encountered, are typically not announced by assistive technology. • Create alert dialog: When an email address is added to the list in the alerts dialog, the change is visible on screen but is not announced programmatically. • Create alert dialog and Login (Register): Dynamic error messages appear when users interact with inputs, for example when deleting data in a field or focusing a field on a form. However, there is no programmatic method in place to alert screen reader users when these messages appear.

technologies without receiving focus.		• Documents and Targets: When zooming in or out (using the buttons onscreen), there is no programmatic feedback to confirm that the action has taken place. • Query builder, Targets and Documents: When interacting with the 'Go to page' field (in dialog), focus is returned to the input after a page number is entered and submitted. There is no programmatic feedback to inform screen reader users that the page content has updated, or which page is currently being displayed. • Query builder: Search results updates are not announced to screen reader users. • My retrosynthesis projects: When tables are sorted, there is no programmatic feedback to inform screen reader users that the sort action has been applied. • Structure editor: When creating a structure from a name, content enters a loading state that may last some time. However, there is no programmatic feedback to inform screen reader users that loading is in progress or when it has completed. • Documents: When interacting with like/dislike buttons, the result of interaction is not communicated to screen readers. • Login (Institutional sign in): The presence of matching results for the institution lookup is not announced to screen readers.
---------------------------------------	--	--

Multimedia

WCAG 2.1 Checkpoint	Conformance Level	Remarks
1.2.1: Audio-only or Video-only (Prerecorded) (A) Provide alternatives for pre-recorded audio-only or video-only content.	Supports (N/A)	There is no pre-recorded audio-only or video-only content.
1.2.2: Captions (Prerecorded) (A) Provide captions for pre-recorded audio	Supports (N/A)	There is no audiovisual content on pages tested.
1.2.3: Audio Description or Media Alternative (Prerecorded) (A) Provide alternatives for pre-recorded	Supports (N/A)	There is no audiovisual content on pages tested.

synchronized audio/video		
1.2.4: Captions (Live) (AA) Provide captions for live audio in synchronized audio/video.	Supports (N/A)	There is no synchronized audio/video content nor live audio.
1.2.5: Audio Description (Prerecorded) (AA) Provide an audio description of pre-recorded video.	Supports (N/A)	There is no audiovisual content on pages tested.
1.4.2: Audio Control (A) Audio can be paused and stopped, or the audio volume can be changed.	Supports (N/A)	No pages feature audio that plays automatically.
2.2.2: Pause, Stop, Hide (A) Users can stop, pause, or hide moving, blinking, scrolling, or auto-updating information.	Supports	There is limited moving, scrolling, or auto-updating information for which the criterion is applicable. Where it is present, moving content respects the browser 'reduced motion' setting.

Usability

WCAG 2.1 Checkpoint	Conformance Level	Remarks
2.2.1: Timing Adjustable (A) Users are warned of time limits shorter than 20 hours and time limits can be turned off or extended	Partially supports	The product does not contain many time limited actions which cannot be controlled by the user. Exceptions: My retrosynthesis projects, Quick search results, Targets and Documents: Status messages appear briefly on screen then disappear. There is currently no way to increase the time limit of these messages or view the messages again once they disappear
2.4.5: Multiple Ways (AA) More than one way is available to navigate to other web pages.	Supports	All pages (except steps/results of a process) may typically be located and accessed in multiple ways. For example, main navigation is consistently available across pages, and secondary navigation is often present in the form of breadcrumb trails representing page hierarchy. Global search functionality that comprehensively indexes content across the site allows users to find specific pages. A sitemap is also available.
3.2.2: On Input (A) Changing the setting of a checkbox, radio	Supports	User input, such as changing the values of form elements, does not initiate unexpected actions or changes in context.

button, or other UI component does not trigger unexpected changes in context.		
3.2.3: Consistent Navigation (AA) Navigation menus are in the same location and order on every web page.	Supports	Navigation menus are consistent across pages. For example, global navigation links in the header are consistent across pages, occurring in the same order; secondary navigation is consistently positioned across appropriate sets of pages.
3.3.4: Error Prevention (Legal, Financial, Data) (AA) For web pages with legal or financial commitments, input can be reviewed and corrected before final submission, and submissions can be reverted.	Supports (N/A)	There are no submissions which require legal or financial commitments.

Mobile User Experience

WCAG 2.1 Checkpoint	Conformance Level	Remarks
1.3.4: Orientation (AA) Content does not restrict its view and operation to a single display orientation, such as portrait or landscape, unless a specific display orientation is essential.	Supports	Pages do not restrict view and operation of content to a single orientation.
2.5.1: Pointer Gestures (A) All functionality that uses multipoint or path-based gestures for operation can be operated with a single pointer without a path-based gesture, unless a multipoint or path-based gesture is essential.	Supports (N/A)	Pages do not utilize or require multipoint or path-based gestures for any functionality.
2.5.2: Pointer Cancellation (A) For functionality that can be operated using a single pointer, at	Supports	All interactive content functions through the Up-Event, allowing users to potentially move their pointer off the component to cancel.

least one of the following is true: • No Down-Event • Abort or Undo • Up Reversal • Essential		
2.5.4: Motion Actuation (A) Functionality that can be operated by device motion or user motion can also be operated by user interface components and responding to the motion can be disabled to prevent accidental actuation, except when: • Supported Interface • Essential	Supports (N/A)	There is no content that utilizes device or user motion.