

# NNELS



## Accessibility Report for Hoopla

October 2021

This report was written with support from the Government of Canada's Social Development Partnerships Program - Disability Component.

The opinions and interpretations in this publication are those of the author and do not necessarily reflect those of the Government of Canada.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#)

## About NNELS

The National Network for Equitable Library Service (NNELS) is a digital public library of ebooks for Canadians with print disabilities and an advocate for an accessible and equitable reading ecosystem for Canadians with print disabilities<sup>1</sup>. NNELS supports principles of openness, inclusion, and choice. NNELS is hosted by the BC Libraries Cooperative, a community service not-for-profit cooperative and a national leader in information and technology services.

Our team of Accessibility Testers has expert knowledge in the areas of accessibility testing, analysis, software development, and leadership. The team works to educate and advise publishers, technology vendors, and public libraries on best practices for accessibility. Our testers have lived experience with a range of print disabilities, including blindness, low vision, and learning disabilities.

---

<sup>1</sup> Print disabilities are defined by Canada's Copyright Act and include visual, mobility, or comprehension impairments such as dyslexia.

## Accessibility Summary

Hoopla has many accessible features in its apps and website, including different visual adjustments and custom help text. However, all accessibility issues in the apps and website need to be fixed to provide a fully accessible user experience.

The recommended accessibility improvements vary between each of the apps and the website. There is one constant across all platforms, however: the labelling needs to be improved to be clear and descriptive for assistive technology users. This includes adding labels to buttons that currently do not have them and fixing verbose labels. This is discussed in more depth in this report, for each version.

Hoopla's video and audiobook players also need to be more accessible for screen-readers and keyboard navigation, so enjoying content is as accessible as possible.

## Introduction

Hoopla provides access to many types of digital content, including audiobooks, comics, ebooks, movies, music, and TV, through participating public libraries. The service is available via a website, mobile applications, or smart devices such as the Apple TV and Amazon Echo.

Our testers used screen-reading and magnification software to assess the usability of the apps and website across supported platforms. A complete list of all the software and operating systems used in this assessment can be found in the Systems and Assistive Technology section of this report. The objective of this assessment is to determine the usability experience of readers with print disabilities and to what extent they can access content through their local public library effectively and efficiently. While this report aims to provide an overview of the accessibility performance across supported platforms, this is not an in-depth review of Hoopla itself, and some functionality may not be discussed at all or in-depth.

The Hoopla website and mobile applications were found to have deliberate accessibility enhancements. These include good use of [ARIA](#) to improve accessibility on the web, custom help text for mobile users, and visual adjustment settings including a dyslexic font. On all platforms—but particularly Android—the app was found to have missing or incorrect button labels as well as other accessibility barriers. These findings—both positive and negative—are documented in the report below, and where appropriate, recommendations have been provided.

## Introduction to Assistive Technology

All mainstream operating systems include built-in screen readers (Narrator on Windows, VoiceOver on Apple devices, and TalkBack on Android) that read the contents of the screen out loud, allowing users with visual disabilities to browse apps and websites, send and receive texts and emails, and accomplish many other tasks with ease. Keyboard commands and custom touch gestures provide a flexible way for a user to find and interact with the

controls on-screen. Windows also has alternative screen-reading software available, most notably a commercial option called Job Access with Speech (JAWS) and a free and open-source option called Non-Visual Desktop Access (NVDA). The text spoken by a screen-reader can be sent to a refreshable braille device. Mainstream operating systems are also equipped with user interface magnification, large text options, and high contrast viewing mode to assist people with low vision.

To ensure usability and accessibility of an application by people with print disabilities, all functions and controls must be accessible using assistive technologies. The DAISY Consortium explains that the basic assumption of accessibility evaluations is that reading systems “should support reading with eyes, ears, and fingers.” (DAISY Consortium, 2017). It should be possible for users to read the content of the document by:

- Reading the text with screen readers or self-voicing text to speech (TTS) applications.
- Adjusting the display including font size, alignment, and color contrast, or combination of some or all of these options.
- Reading the text with a refreshable braille display.
- Reading with assistive technologies designed for persons with dyslexia or other disabilities.
- Reading with the app’s built-in read aloud functions.

## Accessibility Performance and Recommendations

In this section we dive deeper into specific accessibility issues encountered during testing the Hoopla apps and website. Below you will find the testing results and their related recommendations as they pertain to:

- Library Access
  - Signing In
  - Searching, Reading, and Navigating Content
- Visual Adjustments

Finally, the *development recommendations* section contains suggestions for improving the interface on each platform. These suggestions will be relevant to any issues or observations noted above.

### iOS

- Tested iOS Version: 14.4+
- Tested App Version: 4.39.1, 4.4

#### Signing In

All elements on the login screen are easy to locate and identified by VoiceOver. The login button has a label of “Login: Please enter your email and password and try again.” This is

true even if the email and password are both filled in, so this hint is redundant and confusing.

### General

The Hoopla app mirrors the functionality of the website. Bottom tabs allow for easy navigation to the main sections of the app. When the "My Hoopla" tab is selected, there are three sections: "Home," "Borrowed," and "Favorites." When the app is first loaded, a VoiceOver hint tells users to swipe left or right with three fingers to move between these sections. This hint is sometimes interrupted when the app finishes loading and VoiceOver reads the current control instead. These sections can also be accessed with navigation buttons at the top of the screen. VoiceOver does not indicate which of the sections is selected, as it would do with the bottom tabs.

All elements in the settings screen are labelled well, and there is no unnecessary separation of each toggle and its associated text label—in other words, VoiceOver reads the setting name followed immediately by the state of the toggle. This makes navigation much more efficient and less confusing. The two-finger scrub gesture will reliably go back to the previous screen, and a button for dismissing the settings pop-up is always available.

In the recommendation settings, the main category titles are indicated as “section headings.” Changing these to actual headings would allow VoiceOver users to jump quickly from one section to the next using heading navigation. One tester noted that when toggling the selection of a category item, VoiceOver first reads its original state followed by the updated state, perhaps because the app waits for a server response before updating the interface.

### Searching and Browsing

The home screen has several item lists, including “Recommended for You,” “Recent Favorites,” and a “Popular” section for each type of content. These section headers have no control type, so there is no way to efficiently navigate from one section to the next. Tagging these as headings would make this possible. The same list format is used in the "Favorites" section, as well as the "Books," "Music," and "Videos" tabs.

When the “See All” button for one of these categories is pressed, the entire category is shown, with filter and sorting options to further refine the list. When settings are chosen under the filter or sort sections, the button labels do not update to reflect those settings. For instance, selecting a sort option such as “Popularity” will change the sort order of the results, but VoiceOver will not indicate that books are sorted by popularity when reading the sort button. One tester described the experience of filtering titles, including examples:

"Choosing an item such as language causes a list of languages to expand below it. Each language has a name and a number in parentheses indicating the number of titles available in that language. If I double-tap on one of the languages, the list of languages disappears except for the name of the language that I have selected. VoiceOver does not indicate whether the items such as language are expanded or collapsed, and which language is selected. If I double-tap a language again, the list of languages disappears, and the language is no longer listed. I am assuming this means

the language is no longer selected in the language filter. Having VoiceOver indicate which languages are selected when the list is expanded would help to clarify which are currently selected. Also, having status on the filters for User rating, Release Date, etc. to indicate whether they are collapsed or expanded would be helpful."

In a list of ebooks, each title is separated into three controls which VoiceOver users must navigate individually—one for the format (such as an ebook), one for the book title, and the last for its author. This triples the number of gestures required to explore books. VoiceOver navigation would be more efficient if each book listing were combined into a single grouping containing the title, author, and format.

The "Genres and Collections" button appears in the "Books," "Music," and "Videos" tabs. When this button is pressed, the available collection categories are read as both a heading and a button. These "button headings" do nothing when the user double-taps to activate them; instead, they have a rotor action called "Browse the list of all collections." To activate this button, the user needs to swipe down once and then double-tap. This is accessible, but since there is no primary action on the heading (nothing happens when it is double-tapped), being able to open the sub-page directly by double-tapping on the heading would be more intuitive. A VoiceOver hint of "Browse the list of all collections" could then be added instead of the secondary rotor action.

The Details page for a book is quite accessible. When using swipe gestures to move through the screen, the duration or number of pages comes first (depending on whether it is an audiobook or ebook), followed by the publishing date, title, and author. Listing the title and author first would be more logical. The buttons that follow are "Borrow," "Favorite," and "Share." If a book is borrowed, a "Download" or "Resume" button will appear in its place. Using swipe gestures, the order of the three buttons is "Borrow," "Favorite," and "Share." Visually, they are listed as "Favorite," "Borrow," and "Share." Having the swipe order match the on-screen order would make more sense.

After the summary, the author, publisher, and genres of the book are listed. These are not indicated as buttons but choosing them will search Hoopla's catalogue for other items with that author, publisher, or genre. Changing these to buttons would clarify the fact that they can be activated and are not just static text.

The advanced search page uses a filter system similar to the browse screens, but with a few differences: each of the filter categories is labelled with the word "section" with no space before it, such as "Release Datesection." The accessibility labels for these categories indicate when they are expanded, but not when they are collapsed. Changing the label to include the word "collapsed" would help clarify that the controls *can* be expanded and that new content will appear below them. The items underneath each filter are also labelled more clearly, including the number of applicable titles as part of the same control rather than a separate item next to it.

The advanced search screen also has separate edit fields and pickers for the search criteria, including "Title," "Genre," and "Author." When a picker is activated, VoiceOver reads it as a text field with the available options where a keyboard would normally appear. VoiceOver

should not refer to a control as a text field if it does not bring up a keyboard. Using a default iOS picker or an adjustable control would allow a user to swipe up or down through available options, and easily move to the “Done” button.

Search results are rendered as single controls, similar to the “Borrowed” section and unlike the lists of titles in all other sections of the app. VoiceOver users can navigate search results much more efficiently since no extra swipes are required.

The “Borrowed” section has two subsections—one for currently borrowed books and the other for borrowed history. These sections are separated by a control VoiceOver refers to as “section header.” Changing this to a heading would allow VoiceOver users to jump to the correct section efficiently. Unlike the book lists in the “Home” section, each book is represented by a single control, making navigation much more efficient in this part of the app.

### **Audiobooks**

The audiobook player is divided into three sections—the player, the contents and bookmarks, and the details page, which contains “Speed” and “Sleep Timer” controls. These sections can be reached with a three-finger swipe left or right using VoiceOver, and most controls are accessible throughout.

VoiceOver testers reported problems creating a sleep timer: The screen would sometimes become unresponsive after the “Sleep Timer” button was pressed, with VoiceOver reading nothing at all. This did not happen consistently, so after the app was restarted, testers were able to use the otherwise fully accessible controls to set the timer. A two-finger scrub will close this screen, but there’s no visible back or dismiss button.

In the table of contents, some tested audiobooks had no chapter names—likely a result of how the files were named and tagged in the original copy. This is not specifically an accessibility issue but since readers with print disabilities may prefer audiobooks or be unable to read ebooks fluently, good metadata would help such users navigate the sections within an audiobook just as efficiently as those of an ebook.

The “Minimize” button is located at the top of the screen, but in VoiceOver’s swipe order, it is placed after all other controls, so one would need to swipe right through the rest of the screen to reach it that way. It would make more sense if it were at the beginning of the swipe order. When the audiobook player is minimized, it remains at the bottom of the screen. Tapping the name of the book will cause the player to expand once again. A VoiceOver hint on the expand button would be helpful, as users may not realize they can double-tap it to expand the player controls (though this is a common convention in most complex media player apps.)

### **Ebooks**

For visual readers, the ebook viewer provides several useful customizations, including theme, margins, font, line spacing, text alignment, and a choice of scrolling or paginated view.

For screen-reader users, the ebook reader is somewhat usable but has a few significant challenges. Continuous reading works well, and VoiceOver correctly scrolls the text as well as reading any image alt-text that may exist. In-text links (such as a table of contents) correctly jump to the relevant section of text. Scrolling to the beginning or end of a section reveals a “Previous” or “Next” button, respectively. These buttons sometimes require multiple double-taps with VoiceOver before they activate properly.

Once reading begins, the control buttons disappear, and no tester could find a way to bring them back without turning VoiceOver off. Typically, double-tapping with VoiceOver should reveal book controls much like single-tapping would with VoiceOver disabled. The lack of controls makes it difficult to jump to the table of contents, add bookmarks, and perform other basic functions. Turning off VoiceOver and tapping anywhere on the screen will correctly reveal the controls but will jump to the beginning of the current chapter.

VoiceOver’s text selection gestures can be used to select parts of the book, and highlights can be created by triple-tapping and choosing the relevant menu item. It is difficult to find the “Highlight” menu item with either swipe gestures or explore by touch since it appears at a midway point near the top of the screen. When the “Add Note” prompt appears, VoiceOver is unable to find a button to close this prompt, though it eventually disappears on its own.

Comics are not accessible to a totally blind reader and will continue to be inaccessible until image descriptions are added. Beyond that, the comic viewer contains a direct touch area, which is a section of the screen that allows VoiceOver gestures to be passed directly through to the app. This should not be necessary, and there is no instruction for VoiceOver users regarding the gestures that can be used to move around the app. A direct touch area should only be necessary for an app with complex custom control, such as a drawing area or a virtual musical keyboard. In other cases, VoiceOver gestures such as three-finger swipes can be used to pass single-finger swipes through to the app.

### Audio Streaming (music)

On the details page for an album, the durations of each track and the entire album are not shown, even though this information is available on the web. When swiping right from the “Back” button, VoiceOver does not correctly scroll the screen back to the top and will often miss the artist’s name if it has scrolled off the screen. Much like the book details page, it is possible to tap the artist and genre to open a list of relevant albums, but these controls have no type. Changing them to buttons would help users realize they are not simply static text.

The audio player is once again divided into three sections, with the player controls being similar to audiobooks. Unlike the audiobook player, there is no hint instructing VoiceOver users to swipe left and right with three fingers, so this functionality could be easily missed. VoiceOver also does not give any spoken feedback when swiping between these three sections. The minimize button is at the end of swipe order here as well.

The “Repeat” and “Shuffle” buttons have quite verbose VoiceOver labels, such as “shuffle button shuffle off button.” The word “button” is not necessary here since VoiceOver reads

the control type. The label is also somewhat ambiguous—is shuffle off, or will the button *turn* it off? One way to subtly clarify this would be to label the button “shuffle: off.” The colon and resulting VoiceOver pause will indicate to most users that shuffle is turned off at the moment. VoiceOver does immediately read the updated mode after the button is double-tapped.

### Video Streaming (Movies and TV)

The experiences of watching movies and TV are nearly identical, so unless otherwise specified, this information will apply to both.

Much like other details pages, some information can be tapped to initiate a catalogue search for related items. These controls are not indicated as buttons, so VoiceOver users may not know they can be activated. Using headings for cast and other grouped information would also be helpful.

All video controls are accessible with VoiceOver, except for the button which causes iOS to prompt for access to network devices (perhaps related to Chromecast devices). This button has no label at all.

When subtitles are enabled, VoiceOver reads them automatically. Testers noted that the volume of VoiceOver seemed to change rapidly as the video content played. However, it is unclear whether this is a bug or the result of audio processing meant to normalize volume. No titles were found with audio description, so if this feature exists, it was not tested, and it does not seem possible to search for or filter shows based on their inclusion of a description track.

### Visual Adjustment

Hoopla has several visual adjustment options across various sections of the app. For example, the font and many aspects of the text appearance can be adjusted in the ebook reader, but the font and appearance of video captions can also be changed.

Unfortunately, since comics are comprised of images of text rather than the text itself, they have a fixed layout, and the appearance of text can't be changed. However, the app has a magnifying glass that can be moved around to enlarge sections of the comic.

### Development Recommendations

- Change section headers to headings so that VoiceOver users can move between them.
- Add more control types to the buttons on the details page.
- Combine the items in content lists into a single item per-listing, rather than having separate controls for title, artist/author, and format.
- Do not use direct touch in the comic viewer.
- Fix verbose labels on shuffle and repeat buttons.
- Fix VoiceOver freeze in sleep timer setting when moving to the top of the screen.
- Allow VoiceOver users to toggle and access reading controls in the ebook viewer.

## Android

- Tested Android Version: 11
- Tested App Version: 4.5

### Signing In

The sign-in process was easy and accessible. The “Log In” and “Forgot Password” buttons have no control type; these would typically be a button and a link respectively.

### General

The four bottom tabs have no control type; however, TalkBack correctly identifies the current tab as being “selected.”

On the home screen, TalkBack will encounter a scrollable carousel near the top. These items are all unlabeled and have no control type. Depending on verbosity settings (which are set by the user), TalkBack will either read nothing at all or will read their index (such as 3 of 12). Since these items and the containing carousel have no TalkBack labels and no control type, a screen-reader user would not have any context for identifying and using them.

Toggles on the settings screen are identified correctly as switches, but TalkBack does not read their status until one of them is double-tapped. It is, therefore, possible to tell whether a setting is on or off, but only after toggling it.

### Searching and Browsing

The home screen contains various categories—such as popular ebooks—which are each carousels. These function quite well with TalkBack: Each title listing is only a single control, and all important information about the title is read concisely.

Pressing the “More” button on a carousel will open a screen with filter and sort options, as well as a vertical listing of all titles from the given category. The “Filter” and “Sort” buttons are physically located at the top of the screen, but when swiping right to move through controls, TalkBack incorrectly places these buttons between the last title listing and the bottom tabs. As of this writing, the best way to find them is to touch the “Home” tab and swipe left. TalkBack also encounters one other unlabeled button after the top heading on this screen.

Each filter category is read as a piece of static text with no control type and no collapsed or expanded state. Double-tapping a filter category will reveal the available filtering options, though TalkBack will provide no feedback when the additional content has appeared. The options underneath each filter category are correctly reported as radio buttons that are either ticked or unticked. It is not possible to determine whether toggles such as “Exclude Abridged Titles” are enabled or disabled, and double-tapping any of these checkboxes will immediately cause TalkBack to lose its place in the app. There is no visible in-app “Back” button to leave the filter and sort options, but pressing the system back button will accomplish this.

The filters on the advanced search screen work very well with TalkBack. Extra accessibility hints have been added, instructing the user to “double-tap to change this filter.” When the filter options appear, TalkBack’s focus is correctly placed on the first radio button. If another option is chosen, TalkBack focus jumps back to the collapsed filter, which was just changed. This creates an exceptional workflow where the user can systematically modify filters by simply swiping right and double-tapping as needed.

### **Content Management**

The "My Hoopla" tab is divided into two sections – "Borrowed" and "Favorites." The "Favorites" section resembles the home screen, with carousels for each media type. The "Borrowed" section simply lists all current titles, as well as any recently-borrowed items. Each borrowed title has three controls—the title itself, a play button, and an unlabeled button which appears to do nothing when pressed. When adding or removing a title from favorites, an alert is spoken, but the label for the button does not change. When borrowing or renewing a title, the focus is correctly jumped to the confirmation screen.

On the details page for a title, tapping the name of the author or artist will perform a search. Since these controls have no type, a TalkBack user may never know about this functionality. The close button on the details screen is also unlabeled.

### **Audiobooks**

Several buttons within the audiobook player have incorrect or misleading labels. All three of the top buttons are labelled as “Navigate Back.” The buttons to skip by chapter are called "Rewind" and "Fast Forward." TalkBack also detects an unlabeled button in the middle of the screen.

TalkBack testers found they were unable to expand the audio player during background playback. The playback controls could only be revealed by finding the title from the "Borrowed" section and tapping "Resume."

When accessing pop-ups such as the "Sleep Timer" or "Speed" setting, the new controls are loaded with no spoken feedback from TalkBack, and it is necessary to explore by touch to escape the player controls. The "Close" button also has no label, and since pop-ups do not close after an option is selected, it is necessary to find and activate this button before the player can be used again.

### **Ebooks**

Ebooks are presented in a web view which defaults to full-screen. Each chapter is rendered fully, so TalkBack’s continuous reading mode will stop at the end of the chapter. At this point, a button becomes available to move to the next chapter, and reading can continue. TalkBack recognizes all standard HTML, so any links or image descriptions would be rendered correctly. Extensive visual settings are available, including a selection of fonts, line spacing options, preset themes, and text alignment. Combined with the overall usability via TalkBack, it is apparent that significant accessibility enhancements were added to this section of the app.

In contrast to iOS, TalkBack users can double-tap the text to reveal reader controls, though testers found it was often necessary to double-tap a second time. All controls are easy to identify and use. When navigating the various pop-up screens, it is possible to tap the surrounding screen area to return to the book. This is represented by a button that has no label.

The labels for radio buttons on the settings screen are very clear but unnecessarily verbose (for example, “Tap here to apply the dyslexic font”). The accessibility hint would be a better place for this kind of instructional text. When navigating quickly through a screen with dozens of controls, hearing the words “tap here to apply” for every option is a significant disadvantage.

The in-text search functions well. Results are grouped by chapter and tapping a result will cause the containing page to be displayed. TalkBack does not announce this screen change.

### **Audio Streaming (music)**

The music player is intuitive and broadly accessible. The only exception is the button to collapse or expand the track list, which has no label. While the "Shuffle" and "Repeat" buttons and their resulting pop-ups are accessible, the buttons themselves do not change based on the current shuffle or repeat mode.

Much like the audiobook player, the music player cannot be expanded with TalkBack once it has been dismissed, so it is necessary to find the album in the "Borrowed" section and resume playback.

### **Video Streaming (Movies and TV)**

The video player is quite simple and intuitive, with standard buttons and a familiar layout. The playback controls are hidden while a video is playing, but TalkBack treats a section of the screen as a button that will reveal them. Unfortunately, the controls disappear after several seconds, so selections must be made quickly. This can be challenging for a TalkBack user who needs to first explore the buttons one by one.

Subtitles are available on all tested content. These can be visually customized with size and appearance options but cannot be read manually or automatically using TalkBack. When selecting a subtitle language, the word “checked” is merged into the language name, so English becomes “checkedenglish.”

### **Visual Adjustment**

The text size, font, and layout of ebooks can be extensively customized, including an option to synchronize with Android appearance settings. The size and colour of video subtitles can also be modified.

### **Development Recommendations**

- Ensure buttons and other controls have informative accessibility labels.
- Ensure the audio player can be expanded using an accessible control.
- Shorten labels of search filter options and ebook viewer settings.
- Avoid hiding video controls when TalkBack is enabled.

- Ensure toggle switches throughout the app properly indicate their enabled or disabled state.

## Web

### Signing In

When signing up for an account, some of the text fields are not read correctly when using the tab key to move through them. These controls have a label positioned above them but do not have a label attribute attached, so a screen-reader may not identify them properly. It is still possible to manually review the label. If there are any errors in the submitted form, the relevant text fields are not marked in a way screen-readers can detect. Using the aria-invalid attribute on these fields will help users spot and correct mistakes. The sign-up form has an unlabeled button after the “confirm password” field, which is never reachable with the tab key, and which appears to do nothing when pressed.

The list of available libraries is presented as a list of menu items; however, the menu is not navigable with the arrow keys; the tab key will move through each menu item instead. This library list may make more sense as a list of links.

### General

Overall, the Hoopla website is largely accessible across multiple operating systems. Landmarks and headings are used to denote important sections of each page. Most controls are labelled well. Controls in settings screens do not indicate their status, so a screen-reader cannot determine whether an option is enabled or disabled.

The “My Hoopla” and “Browse” submenus will auto-expand when they gain keyboard focus. This poses a problem for keyboard users who navigate with the tab key since they will need to tab through every submenu item before reaching the rest of the page. In some cases, a screen-reader will also move keyboard focus as the user navigates the page, automatically causing the menus to expand. A common solution to this problem is implementing arrow-key navigation and removing the tabindex from the submenu items altogether. Removing the auto-expand behaviour from keyboard focus would also be helpful since it is already possible to expand or collapse them by pressing enter or ESC. Lastly, screen-readers do not report the expanded or collapsed state of these menus. This can be fixed by setting aria-expanded appropriately as the menu expands and collapses.

Other menus on the page also lack arrow-key navigation and must be tabbed through. This includes the search category selection, the trending searches list, and the library selection when signing up for an account.

### Searching and Browsing

On the “Home” and “Browse” pages, the carousel for promoted items is unusable with NVDA. This is because each item has an aria-label of “Promo for” and despite the image having alt-text, NVDA only reads the label for the containing item. Other screen-readers handle this properly. Pressing the tab key will also cycle through all of the items in this carousel. Promoted items within the curated lists are similarly mislabelled, and there is no

separation between the curated and promoted items. One possibility is to simply add the word “promoted” to the beginning of the label.

Title listings have been given a custom label for screen-readers. This applies to search results as well as the lists of titles on the homepage and in browse screens. For example, "Best Of The Mamas & The Papas - Superstar Series,by, The Mamas & The Papas,MUSIC,Available,." The overuse of commas suggests that these labels were engineered to be spoken clearly, with appropriate pauses. In reality, some users may configure their screen-reader to read all punctuation, which would cause each comma to be read as the word “comma,” and others may use a braille display, so modifying the output this way is not recommended. Instead, consider the following changes which would work well-using speech and braille:

- Remove all commas and make sure all information is separated with a space.
- Only include availability status if the title is unavailable. In that case, include it in parentheses at the beginning so that users will know immediately.
- In lists of titles where the format is known to be the same, exclude it. In lists of titles with varying formats, include it at the beginning since it is the most relevant piece of information about the title.

Example of the new output: "(unavailable) Music: Best of The Mamas & The Papas - Superstar Series by The Mamas & The Papas."

When a search is performed, many filters appear above the search results. These include format, user rating, release date, date added, and language. Whether they are expanded or collapsed, these filters appear as static text (no control type, and no expanded state.) A screen-reader user may not identify these as filters or realize they can be expanded, though NVDA does identify them as clickable if the correct setting is enabled. Changing their role to “link” and using aria-expanded would go a long way toward making these controls more accessible. Once these filters have been expanded and collapsed, each of their names appears twice to a screen-reader.

Once filters have been applied, a button to remove all filters becomes available. This button does not react to the space bar or enter key in NVDA’s browse mode and does not react to the space bar in focus mode either. The only way to clear filters is to tab to the button and press enter while in focus mode. This is seemingly because the clear button is a nonstandard control that is given an ARIA role of “button” for easier identification. It accepts the enter key and a mouse click, but no other activation methods have been implemented.

The filter options have great aria labels, but they could be more concise. For instance, a radio button called “Apply 4 stars and up rating filter” could simply be called “4 stars and up.” Screen-reader users will know that it is a filter because it is under the filters section and that it is a rating filter because it’s under the “user rating” subsection.

When the "Borrow" button is activated on a title, a short message appears with the borrowed time and a confirm button (also called "Borrow"). For a screen-reader user, there

is no warning that this message has appeared. One possible solution is to use a dialog element around the alert and moving the keyboard focus to the new dialog when it appears.

It does not appear possible to rate a title using the keyboard. The button to do so appears as a clickable element but cannot be reached with the tab key. This may be another control that only responds to direct keyboard input, but the lack of a tab stop makes it impossible to set keyboard focus to the button. Additionally, the current rating is shown as an unlabelled graphic that is not readable by any screen-reader.

### **Audiobooks and Music**

The audio player appears whenever the play button is pressed on a book or album, allowing the user to navigate Hoopla without stopping playback. All controls have good labels, and the player is intuitive and responsive.

In screen-reader order, the player controls are at the bottom of the page below the footer. Regardless of whether it is collapsed or expanded, the button to toggle its state is always called "Expand player." Since there is no feedback when it is pressed, a screen-reader user would need to navigate past the button to determine that the player is collapsed. It would also be helpful to have a heading before the player controls.

When buttons such as "Sleep Timer," "Chapter Menu," "Bookmarks Menu," and "Playback Speed" are pressed, the pop-up appears with no warning to the user. In screen-reader order, the new controls are at the bottom of the page below all other player controls. These pop-ups may be better rendered in a dialogue that closes after a selection is made, dropping the user back to the originating button. Similarly, when bookmarks are added, there is no feedback to indicate this has been done, but a message can be seen at the bottom of the page. An aria-alert would cause the screen-reader to speak a confirmation message.

When reading an audiobook, the readouts for elapsed and remaining time are read with no separation, such as "3:1026:50." When navigating the seek slider for a book with a screen-reader, it appears to read the number of elapsed seconds since the beginning of the book, so in later chapters, this number will always be high and relatively meaningless. A percentage of the current audio file would be helpful.

### **Ebooks**

Ebooks are read using an integrated book viewer which presents the text by chapter. Continuous reading with a screen-reader is possible until the end of the chapter is reached. Many visual settings are available, including brightness, bold text, and a font selection which includes an option simply called "Dyslexic." Unlike the audio player, choosing a menu option such as "Chapters" will correctly switch keyboard focus to the back button at the beginning of the pop-up. This is a relatively minor detail with a significant impact on efficient navigation. Having pop-ups in a separate dialogue would still be preferable since screen-reader focus could never leave the pop-up.

When the text has been visually scrolled to the end of the chapter, a "Next Chapter" button appears. Since NVDA does not visually scroll the text consistently, this button never

appears, and it is necessary to continuously turn pages until the next chapter is reached or access the chapters menu. One possible solution would be to insert the chapter navigation buttons at the beginning and end of the text rather than dynamically generating them based on the reading position. They would become visible after the last page is turned, and screen-reader users would always be able to find them.

When moving to a search result within the current book, a screen-reader receives no indication that new text has loaded, and since the entire chapter is visible, the search result might be pages away from the beginning. Most screen-readers have built-in find functions, so this problem is avoidable if it cannot be fixed, but moving keyboard focus directly to the search result or the relevant page would be helpful.

### **Video Streaming (Movies and TV)**

On the details page for a TV show, the episode number, and duration get merged without a space, causing NVDA to combine the numbers ("EPISODE 144 MIN").

Unfortunately, none of the buttons in the video player are labelled. These include the "Play/Pause" button and the "Rewind" and "Forward" buttons, which means screen-reader users may have trouble accessing the essential functions of the player. Keyboard focus is also occasionally reset to the top of the video player, so navigating this interface with a screen-reader is challenging.

Subtitles are available for most—if not all—video content. On iOS, these are read automatically by the VoiceOver screen-reader, but testers found that the desktop site had no such functionality, making non-visual access to subtitles much more difficult. The visual appearance of text can be adjusted, which is a nice touch.

The seek slider responds to the arrow and page up/down keys; however, keyboard adjustments are slow because the slider seems to jump back to its original position until the chosen part of the video loads. For instance, when pressing the page down to jump from 10 to 20 seconds, the seek bar briefly changes to 20, jumps back to 10, and then settles on 20. Rapid presses of the page down key will simply continue jumping between 10 and 20 seconds.

Much like the iOS app, video content has subtitles available but no audio description was found. However, the settings screen has an "Audio" setting which is simply set to "Default" on all tested video content. This may imply that audio description is available on some titles, but the setting might also be used to switch audio languages when others are available.

### **Visual Adjustment**

The website itself does not appear to have visual adjustments. More settings to adjust the interface appearance—such as enlarged controls and text size—would be a useful addition for low-vision users.

The ebook viewer has several settings to control text appearance. This includes a dyslexic font amongst others, margin and line spacing adjustments, and a selection of themes. The text of subtitles can be similarly adjusted.

Comics are image-based and cannot be adjusted dynamically, but a movable magnifying glass is available, and low-vision testers found this feature helped them read the text more easily. Having the ability to adjust the magnification level would be helpful.

### Development Recommendations

- Change settings controls so that their state is indicated by screen-readers (radio buttons).
- Properly indicate whether menus are collapsed or expanded using ARIA and avoid automatically expanding them when they gain keyboard focus.
- Fix labels of promoted items so that they include the name of the item.
- Remove comma separation from screen-reader labels for title listings.
- Properly indicate whether filters are expanded or collapsed, or consider using combo boxes instead.
- Ensure any pop-ups are contained in a dialogue that gains keyboard focus, or if they are auto-closing alerts that require no action from the user, are spoken by the screen-reader using an ARIA alert. This includes the borrow confirmation as well as pop-ups within the audio player, such as "Sleep Timer."
- Ensure titles can be rated with the keyboard.
- Ensure the "Expand" button properly indicates when the audio player is already expanded, either by changing the label to "collapse" or by using ARIA.
- Ensure the seek slider for audiobooks indicates the time or percentage of the current chapter, rather than the elapsed number of seconds since the start of the book.
- In ebooks, ensure a screen-reader user can always move to the next or previous chapter even if the correct portion of the text is not visible.
- Move screen-reader focus directly to text search results, or at least to the start of the page where the search text was found.
- Ensure all controls in the video player are labelled.
- Ensure the seek slider for videos can be quickly adjusted with the keyboard.
- If audio description is available, allow filtering video content based on whether it has audio description.

### Conclusion

Testers found that the Hoopla platform has many accessibility enhancements across all operating systems, but each version of the app has minor and moderate accessibility barriers as well. Technically inclined screen-reader users can work around most of these barriers, but others may find themselves unable to access certain functions and content. For low-vision and dyslexic readers, the app provides an unprecedented array of visual adjustments in ebooks and many options for video subtitles as well. NNELS appreciates the accessibility enhancements across all tested platforms and hopes that the suggestions in this report can assist the developers in creating an even more accessible Hoopla.

## Systems and Assistive Technology

- Operating Systems
  - Windows 10 (20H2)
  - macOS 11
  - iOS 14.4+
  - Android 11
- Mobile Applications
  - Hoopla 4.39.1 and 4.4 (iOS)
  - Hoopla 4.5 (Android)
- Browsers
  - Chrome 89
  - Firefox 87
  - Safari 14
- Screen-readers
  - NVDA 2020.4 (Windows)
  - JAWS 2021 (Windows)
  - VoiceOver (MacOS)
  - VoiceOver Touch (iOS)
  - TalkBack (Android)
- Magnification
  - Magnifier (Windows)
  - ZoomText 2020 (Windows)
  - Zoom (iOS)

## Acknowledgements

The following testers and editors contributed to this report:

- Karoline Bourdeau
- Leah Brochu
- Mélissa Castilloux
- Maryse Glaude-Beaulieu
- Simon Jaeger
- David Kopman
- Riane LaPaire
- Ka Li
- Richard Marion
- Laetitia Mfamobani
- Deanna Ng
- Megan Sellmer
- John Ylioja

Published by the [National Network for Equitable Library Service \(NNELS\)](#), Vancouver BC, 2021