Accessibility Testing of PressReader

Conducted by the National Network for Equitable Library Service (NNELS)

Report date: July 2019

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 authors and do not necessarily reflect those of the Government of Canada.

Published by the National Network for Equitable Library Service (NNELS), Vancouver BC, July 2019

About NNELS:

The National Network for Equitable Library Service (NNELS) is a digital public library of books for Canadians with print disabilities, and an advocate for an accessible and equitable reading ecosystem for Canadians with print disabilities. 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.

Table of Contents

Summary 3

Introduction 3

Summary of Accessibility Issues Across All Platforms 5

iOS with VoiceOver 5

Android with TalkBack 6

Web Browsers with Screen Readers 7

Testing Approach 8

Systems and Assistive Technology 9

PressReader Application Versions 9

Devices, operating systems and assistive technology 9

Libraries 9

Notes 10

Testing Results and Recommendations 10

Library Access 10

Signing In 10

Searches 13

Downloading Titles 15

Reading and Navigating Content 16

iOS with VoiceOver 16

Android with TalkBack 18

Visual Adjustments 21

iOS 21

Android 21

The PressReader Website 22

Windows Browser: with JAWS or NVDA 22

Mac OS Safari with VoiceOver 25

Chrome OS with ChromeVox Screen Reader 26

Conclusion 27

Contributors 27

Appendix A: Evaluation Questions 28

Summary

Readers with print disabilities utilize a variety of assistive technologies to access various applications on their tablets, smartphones, and computers. This includes software or hardware with features that are specifically designed to help people with a wide range of disabilities, such as screen readers and screen magnification software. For this report, our team of testers evaluated the accessibility of the PressReader platform for people with print disabilities and found that it presents several barriers for users of assistive technologies.

While our testers could eventually read magazines and newspapers through the mobile apps and website, this was only after using time consuming workarounds (such as sighted assistance to change to text-view.) This led to more time spent on figuring out solutions ending with user frustration. It is our conclusion that too much time spent on trouble shooting is not an ideal experience for any user, but with applying recommended modifications to PressReader, the accessibility and user experience will improve.

This report highlights the accessibility barriers of the PressReader mobile apps and website; we also explain why these issues are problematic, and advance some recommendations to enhance the usability experience for readers with print disabilities.

Introduction

The PressReader platform delivers an endless stream of top news stories right to the user. PressReader offers premium news content without ads or incomplete story snippets, and aggregates multiple sources from around the world. For this report our testers conducted an accessibility assessment of the app. The following report presents the results of this assessment, and highlights the accessibility barriers found in each of the platforms tested. It aims to give librarians useful information to make recommendations to their patrons, as well as to provide feedback to the vendor, so they can make software enhancements to the mobile apps and website to improve its accessibility.

The objective of testing the PressReader platform is to assess the usability experience of readers with print disabilities,[1] specifically whether they can access newspapers effectively and efficiently through PressReader. Readers with visual or learning disabilities use screen readers[2], refreshable braille displays[3] and other assistive technologies in their computers or mobile devices to access information. It is important that readers with print disabilities can choose the reading systems that offer the accessibility features they require.

[1] Print disabilities are defined by Canada's Copyright Act and include visual, mobility, or comprehension impairments such as dyslexia.

[2] A screen Reader is software that runs at the same time as other programs in a computer or mobile device and reads aloud the text that is displayed on the screen.

[3] A Braille Display is a hardware device that connects to a computer or mobile device and converts text into Braille in real time. It contains sets of pins which are raised and lowered to form the Braille encoding.

iOS and Android devices are equipped with screen readers (VoiceOver for iOS and TalkBack for Android) that read the contents of a smartphone's or tablet's screen out loud, which allows users with visual disabilities to browse apps, open links, send and receive texts and emails, and accomplish many other tasks with ease; to open an app or activate a function, users tap on the icon twice; to move through the screen users navigate by swiping, as well as using other gestures to allow them to read text, type, and perform all other tasks. These devices are also equipped with an on-screen magnifier, large text option, and high-contrast viewing mode to assist people with low vision. For accessing laptops or desktop computers, screen-reading software helps blind people to navigate and interact through a variety of keyboard shortcuts, which makes it possible to perform these tasks without using a mouse. It is important that websites and phone applications take these needs into consideration in the design process to ensure optimal usability for all, including people with print disabilities.

In order 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, and/or combination of multiple display adjustments
- 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

The PressReader app was tested on iOS and Android mobile devices; testers also assessed the PressReader website in Windows and Mac computers using a variety of assistive technologies including screen readers and refreshable Braille displays. A complete list of platforms and technologies can be found in the Systems and Assistive Technology section of this report. Our main focus is on the priority issues prevalent across all platforms, as well as platform specific issues.

Information on the iOS and Android apps is presented first, organized by sections grouping similar features. We highlight the problematic issues as they relate to the different functions of the application in each of the platforms tested, but we do not elaborate on every single feature. You can find the entire list of questions in Appendix A. We also include information on the usability of the PressReader website on Windows, Safari and Chrome OS. Since many of the findings related to the websites are similar across the different sections, we describe these issues more broadly, and present specific examples to illustrate some of the barriers relevant to each operating system.

Summary of Accessibility Issues Across All Platforms

The PressReader application for iOS and Android can be used by readers with print disabilities relying on screen readers in their mobile devices through their local public library. Both apps, however, present several accessibility issues that our testers identified as barriers for an average user. While the amount of access to newspaper and magazine articles is impressive, testers could not easily navigate the apps and website due to the many unlabeled controls and buttons, as well as the complex interface.

The PressReader mobile apps and website need changes to the development process and interface to improve the user experience for people relying on assistive technologies. The most important priorities are:

- Label buttons and fix other element labels. Ensure that all buttons in the app (for both iOS and Android, as well as in the website) are labeled with text. Unlabeled buttons are not easily identifiable by people who rely on screen readers.
- Design the size of buttons and tabs to be large enough for gesture interaction. Throughout the mobile apps, many of the buttons are quite small; the smaller buttons are the more difficult it is for a user to find them when exploring the screen by touch.
- Have publications open in text view by default if a screen reader is active, or at least offer the option to set text-view as the default in the app settings.
- Simplify the structure of the website with a logical use of cascading headings to improve the ability of screen reader users to navigate the website and find content.

iOS with VoiceOver

Testers accessing the app on iOS mobile devices used VoiceOver, the native Apple screen reader, which uses many gestures. VoiceOver reads elements in a logical linear manner, from the top left corner to the bottom right. Swiping right moves the focus to the next element in the user interface and swiping left moves to the previous element. The entire app and all its menus should be accessible by these swiping gestures.

The testers using iOS with VoiceOver found that the PressReader interface should be more predictable and consistent for blind people to find it usable. One way to help this is to improve navigation for users with print disabilities. The app needs a labeled navigation button to move back and forth from every screen, and/or the navigation tabs at the bottom of the screen need to be consistent on every page, and reappear as the user navigates through the app. These navigation buttons also have to be large enough to interact with gestures such as tapping; as it is now, they are too small.

Testers also found that it was easy to get stuck in the app, as VoiceOver often could not see a back button. This meant that the only way to exit from these situations was to close and reopen the app. This issue reflects the non swipe-able architecture of the app overall, which doesn’t allow for smooth navigation.

It is rather common to hear VoiceOver struggle with what we refer to as “backfield interference”, when the screen reader reads text and controls that are no longer visible on the screen. When accessing a menu or dismissing a popup, VoiceOver often reads controls that are not displayed or currently inactive. Instead of reading the active screen, it reads the fields that were accessed previously. Most often this occurs at the end of screens (the user swipes beyond the final control,) but sometimes it occurs throughout.

The unlabeled buttons also create a barrier for navigation. When launching PressReader, a browse section appears at the bottom of the screen. When the user navigates through the section by swiping, VoiceOver just reads “popups html image landmark.” This gives no clear indication to what the section is for, and makes it difficult to navigate through the app. While the user can locate a few more buttons, such as “Select Publication” and “Follow” button, there is no way for them to determine the names of the magazines/newspapers on that section.

Accessibility priorities

The following is a list of what the testers identified as top priorities for recommended improvements for iOS with Voice Over.

- Label all buttons and use meaningful names for them. There are a lot of mislabeled and several unlabeled buttons. Most mislabeled buttons have their icon name instead of a proper accessibility label. Specifically, many publication names have an “i arrow” button as well as a completely unlabeled button next to them. Most buttons start with “i” or “ic.” This applies anywhere there are lists of publications, including the home screen and search results.
- Have publications open in text view by default if a screen reader is active, or at least offer the option to set text-view as the default in the app settings.
- Ensure that screen reader users don’t get stuck in the app, because of inaccessible back buttons and no way to make the navigation tabs at the bottom of the screen reappear.
- Improve the read aloud feature (activated by hitting “Listen” under the “i Customize” menu) so that it starts reading at the beginning of publications, and is an easier feature to find.
- Fix the screen refreshing bug when downloading.

Android with TalkBack

When accessing the app in Android mobile devices, TalkBack is used as a screen reader. The testers using Android with TalkBack found that the app is difficult to use with adaptive technology, and its counter-intuitive design is, as one tester put it: “tremendously frustrating to interact with.” To improve the user experience, it is critical that hidden text be removed, or properly hidden, so everything read by TalkBack is actually visible and clickable. The app has a complex design, so it is crucial that all controls be labeled and properly designated as standard, clickable items. Overall, it would be helpful to include captioning cover images, using standard and clickable buttons, labeling edit boxes, and offering tooltips to decrease the learning curve.

As was touched upon in the iOS section, the Android testers also found numerous unlabeled controls, many of which are buttons. For example, on the home feed section, the button to pull up options for a recommended article is completely unlabeled; this includes the options such as “Translate,” “Save,” and “Listen.” When the user swipes this area, TalkBack just says “unlabeled button.” This means that this feature is inaccessible to most readers with print disabilities.

The publications tab also has many blank elements; two examples include the areas for “top magazines” and “see all.” The navigation “X” button for back/close is also unlabeled. With so many unlabeled or mislabeled controls it is impossible for the user to know where they are as they move through the app. For instance, if the user keeps swiping to the right to move to another area or page, TalkBack remains silent and only plays the sound indicating that the area or page has moved without actually informing the user of where they are in the app.

Another issue that the testers discovered is that when the main app page is opened, the screen reader gets overwhelmed by the slider and just reads the number of titles for each page of the slider, and restarts the list midway each time the slider moves. This results in the user being caught in a loop of the screen reader saying larger numbers over and over again. This loop is deeply distracting, and only stops when the user moves to another page.

Accessibility priorities

The following is a list of what the testers identified as top priorities for recommended improvements for Android with TalkBack.

- Make sure all buttons, menus and controls are properly labelled.
- Make text readable with TalkBack, allow default view to be text view.
- Position the “Listen” button more prominently, as this is the only way of accessing the publication’s content with a screen reader.
- Ensure that all elements are focus able and that the reading order is logical.
- Ensure that a magazine stops playing when it is closed.
- Adjust the settings on the main app page so the screen reader is not stuck reading out the slider in an endless loop of numbers.

Web Browsers with Screen Readers

When accessing the platform through the website in a Windows browser, Safari or Chrome, readers with visual disabilities use screen reading software (JAWS/NVDA, VoiceOver, and ChromeVox respectively). For more details on the systems and technology used in this report please refer to the section on Systems and Assistive Technology (page 9.) Many of the findings related to the websites are similar across the different platforms, so we describe these issues more broadly in this section.

The PressReader website presents significant accessibility barriers for most screen reader users to successfully read articles in it because many links cannot be activated through the keyboard. The testers also found that several text labels for buttons and list items were either unlabeled, labeled incorrectly, or provided insufficient information to help the user know what the button or item is for. For example, at the bottom of every page there are a huge number of headings that simply read as "English" by the screen reader. This was found to be true regardless of the browser and screen reader combination. Clicking these has no effect; testers could not interact with them, and had no idea what these headings are and why they are there.

Windows testers found that all the content available through the website was broken up by hyphens (ascii 173 characters) throughout, often occurring in the middle of words. This causes the screen reader to mispronounce everything, with a negative impact on reading comprehension and their usability experience that is likely to discourage average users simply trying to read news on the website, or through the mobile apps.

Accessibility priorities

The following is a list of what the testers identified as top priorities for recommended improvements for accessing the platform through a browser.

- Make sure all buttons, menus and controls have meaningful labels and ensure that they can be activated through the keyboard.
- Simplify the website structure; use headings that cascade to separate sections on the website and enhance navigation.
- Use popup dialogs with progress bars to give audible feedback via the screen reader of what is going on, such as opening or downloading a magazine; avoid using a silent sidebar.
- Fix hyphenation issue in text articles in Windows.

Testing Approach

NNELS developed a list of criteria for testers to perform a structured review of the accessibility of mainstream library reading applications. We drew on the guidelines that the DAISY Consortium developed for the systematic assessment of the accessibility of hardware and software-based reading systems. This methodology helps to determine the accessibility of the functions and features of each reading application from the user perspective.

Six testers with print disabilities (including visual impairments and learning disabilities,) accessibility expertise, and extensive experience using assistive technology, reviewed the PressReader mobile applications using screen readers. One additional tester who is sighted and proficient in the use of the screen reader in her Android phone also assessed the application in the Android platform; in particular, those features related to visual adjustments aimed to provide access to readers with print disabilities other than blindness, such as low vision or learning disabilities.

All testers performed several tasks, corresponding to the different functions of the application, and answered questions systematically. The questions for all functions are grouped into various categories, including:

- Library access – Creating an account, login, searches, results and downloads
- Reading – Screen reader resuming at the same position; pausing before different sections; reading image descriptions; correct pronunciation of words in other languages; text to speech indicating emphasis
- Navigation – Moving through different elements in an article such as headings, paragraphs, as well as spelling words
- Visual adjustments – adjusting colour contrast, font sizes and types, or margins

A full list of questions can be found in Appendix A.

Systems and Assistive Technology

PressReader Application Versions

- iOS versions 5.3.8 and 5.4.1
- Android version 5.3.18 and 5.4.19

Devices, operating systems and assistive technology

- iPhone and iPad with iOS 12.01 using VoiceOver screen reader
- Tablet and smartphone with Android 7.1.1 running TalkBack 6.2.0.19 screen reader
- Mac OS version 10.14 Mojave with VoiceOver screen reader Version 10.14
- Windows 10 Home (x64) with the following screen readers:
    - JAWS for Windows versions 18.05038 and 2019
    - NVDA version 2018.3
- Windows Browsers
    - Internet Explorer version 11
    - Chrome Version 72.0
    - Firefox 66.0
- Mac OS browser
    - Safari 12.1
- Chrome OS (Cloudready) with ChromeVox screen reader
- Braille displays
    - Braille Edge 40
    - BrailleSense U2-Mini

Libraries

Testers accessed the PressReader app through their local public libraries, including:

- Edmonton Public Library
- Markham Public Library
- Richmond Public Library
- Toronto Public Library
- Vancouver Island Regional Library

Notes

The results of testing included in this report were accurate at the time of delivery/publication. However, they may not reflect all personal experiences, which can be affected by the version of the application, as well as the assistive technologies used. In the following pages, we provide detailed information related to the most prevalent accessibility issues as they pertain to the user experience, but this report does not include descriptions of all features of the PressReader mobile apps and website.

Testing Results and Recommendations

In this section we dive deeper into specific accessibility issues of the different features of the PressReader app and website. Below you will find the testing results and their related recommendations as they pertain to:

- Library Access
    - Signing In
    - Searches
    - Downloading Titles
- Reading and Navigating Content
- Visual Adjustments
- The PressReader Website

We highlight some of the problematic issues as they relate to the different parts of the application in each of the platforms tested; but we do not elaborate on each of the features. To see the entire list of questions, please refer to Appendix A.

Library Access

For this evaluation, Library Access includes all features related to accessing PressReader content through a user's public library, such as signing-in, searching for articles and publications (including filtering searches,) downloading and removing articles, and setting preferences.

Signing In

If publications are offered based on the service a user is signed into, an audible indication of whether or not the user is signed in is necessary. As it is now, users can search, filter and select and only then realize they need to sign in. We recommend adding an audible indication for this feature when a user first opens the app to avoid the confusion of discovering this requirement after searching for and selecting a title.

Some testers reported that the login process was confusing. It must be done through a browser and the respective public library's website, but it became apparent as we went through our testing that some libraries offer clearer instructions than others. Below we go into more detail on the sign in issues for both iOS and Android.

iOS with VoiceOver

The log-in and sign-up process presents some barriers, which testers could not overcome. A main barrier is that they were required to sign up and link their library account on the PressReader website.

When the user navigates to the sign-in options and chooses library (labelled as “ic Library”) they get to the screen where they have to choose their library and enter their library card number/pin. It is possible to type the card number and pin fine with the keyboard or use dictation since they are standard edit fields. However, when using dictation the app enters an extra space before and after the inserted text, which provides an error even though the user has entered the two pieces of information correctly.

Recommendation. Enhance these edit fields to be stricter with data validation.

Another issue our testers found is that, after finishing typing in the card number, tapping on the return key does not take the user to the next field, which usually works with other apps.

Recommendation. Instead of labelling the button as “Done” it may make more sense to provide a “Next” button that would take the user to the next field.

Testers noted that the “Select Library or Group” button is not indicated as a clickable element with VoiceOver. After navigating to the button, it just says “Select Library or Group” with no audible prompt to let the user know it is a useable button.

Recommendation. Ensure that VoiceOver can identify that a button or link has been selected and is clickable.

The option to search for the library is not swipe-able. After clicking the “Select Library or Group” button, the app asks for the user’s location, but the following screen with the map is inaccessible for users relying on VoiceOver. Testers noted that it is possible to look at the layout of the map by exploring the screen with their finger, but it is not possible to find any way to select the location in the map.

Recommendation. Ensure that options are swipe-able for VoiceOver and can be easily selected by users with print disabilities.

If the user gets to the screen with an edit field followed by the list of libraries, they can type in their city and pick the library, but it is not possible to know where the list of options is since the screen refreshes. Looking through the options, it appears to be the same screen as the initial library sign-in screen with the pin/card number. It is only until after navigating down the page using VoiceOver gestures that the user realizes they still have to click on the library. This confusion may stem from the fact that only one library is listed at a time, and once the library is selected the first portion of that page looks the same.

Recommendation. Use a heading to separate the library search results.

After the user enters all the information, they are taken to the next screen, which asks them to create an account. Although this screen is mostly accessible, the only problem is that when the user types in a name/password and clicks next to finish signing up, the next screen only shows a heading and a back button. If there are any other messages, they are completely invisible to VoiceOver.

Recommendation. Ensure all messages on pages are readable for screen readers, and that all controls and options are accessible.

Things take longer with a screen reader, and having to go through all the steps as described above would take a visually impaired user significantly longer than a fully sighted one. This can lead to user frustration if there are any additional accessibility barriers. We recommend that the process to sign up be simplified by fixing these issues as described. This would improve functionality to sign in with a library card within the app and lessen user frustration for those with visual impairments.

Android with TalkBack

Testers noted there were mislabeled buttons during the sign in process. The button to sign in is labeled "Collections" as displayed visually, but the button reads with TalkBack as "Bookmarks."

Recommendation. Use correct and meaningful button labels.

It is possible to select the library and enter the library card number, but it is not possible for users relying on screen readers to enter their own pin without sighted assistance, as the password box doesn't have the option to show characters as the user types. This means that TalkBack reads all characters as "full stop" and "bullet" when they are selected. An email address and password are also required by the Agreement step, but again, it was impossible to know which character is being entered into the password field.

Recommendation. Allow for the password box to have the option to show characters so users know what they are typing.

After the tester signed in, the Agree screen pulled their name and showed both last and first names in the first name box. It then failed to submit the sign in information until the user removed their last name from the first name box and typed it into the last name box.

Recommendation. Ensure that the user can enter their information into each edit box as needed, and do not allow for any auto complete.

When one tester opened a magazine a couple of weeks later, the app prompted him to select a library and sign in again. In order to sign in again the process was presented as through the web view, but the "Select Library or Group" button don't read as a clickable item. Another issue is the screen does not wrap properly when swiping to the right; one must swipe to the left to even find this control; furthermore, the button does not respond consistently. The tester had to tap the button three times before it would activate. As well, the edit box to search for a library is unlabeled; TalkBack just says "Edit Box."

Recommendation. Make sure all buttons are labeled correctly and that the screen wraps properly for swiping.

Most of the controls on the Select Library screen that TalkBack indicates as available aren't actually displayed on the screen. This appears to happen when the app paints new elements over pre-existing controls from the previous screen. This is highly confusing for users of assistive technology; it is important that apps present reliable information to screen readers.

Recommendation. Make sure that each new screen presents its own elements to avoid overlapping.

After entering the name of the local library, the user has to tap the list of results and swipe up to the correct library. Although the library name appeared in the pane below the search results, clicking its name or branch canceled the process. Correctly clicking the library name in the list returns the user to the Sign In through web view, where they must once again swipe to the left, and past a few unlabeled controls to find an unlabeled required edit box; as indicated by a list above, this is the place for the library card number.

Recommendation. Ensure all controls and edit boxes are clearly labeled and ordered.

The labels used to log-in are difficult to understand and cause confusion. At the bottom of the screen, there is a control to bring up the main menu that contains items such as help and settings, and each item has the word "navigation" in front (e.g. the item settings is labelled as "navigationsettings.") The same is true with the popup menus; each item has the word "popup" in front. Also, the edit fields to login had labels such as "authorization username" and "authorization password."

Recommendation. Use simple and meaningful labels for menus, buttons and controls. For example, the labels should just say "username" or "password."

By applying these recommendations it will make the sign-up and log-in process less confusing and easier to navigate.

Searches

This applies to all content types – newspapers and magazines.

iOS with VoiceOver

The search function only allows the user to type in one letter; while this is enough to locate the newspaper/magazine, it is not ideal because the user cannot narrow their search even more, which is very important if there are many search results.

Recommendation. Ideally, users should be able to type in an entire phrase.

In the list of search results, there is an unlabeled button prior to each newspaper or magazine. This button contains the name of the newspaper/magazine and the label itself is on the next element. There is a similar issue on the publications screen, which has a few unlabeled buttons. When the user navigates past the search and filter to the newspapers, they find this unlabeled button beside each newspaper, which for example could have the name of "Toronto Star," but the actual label is on a separate element when swiping right with VoiceOver.

Recommendation. These two items (label and button) should be combined into one element.

The testers also experienced that while there are categories on the home screen, it is hard to find them because there are no headings or other meaningful separators between them. Once the user finds the desired category, it is possible to navigate the publications inside it.

Recommendation. Create headings or other meaningful separators between categories.

By applying these recommendations it will make searches less confusing and easier to navigate.

Android with TalkBack

The main screen of PressReader opens into a long list of publications, sorted by category, country and province. Unfortunately, each time a filter is selected, focus jumps back to the top of the screen, so the user has to swipe all the way from the top of the screen back down to the next filter. Testers also indicated that the multiple navigation controls on the side frequently get in the way of swiping down to the content.

Recommendation. Navigation controls could be moved to a menu, slid down to the bottom of the screen, or be activated by a tap of a Navigation button, so that the user can find the actual content easily.

The testers found that the filter option is challenging to use. Users can see the different categories, such as language, only after they enter in a query. This is unintuitive and causes confusion for the user as they attempt their search. When the tester tried to navigate the results, they encountered a table with blank elements.

Recommendation. Have the filter option available and clearly labeled in the search area of the app.

There are issues with the screen reader focusing on the different elements. Normally one swipes to the right to explore what is on the screen; when one swipes to the left, it would show something that the user has already seen. In the PressReader app, however, when swiping left the tester found elements on the screen that they have not seen yet.

Recommendation. Ensure that the different elements can be accessed by swiping, and that the order is consistent.

The list of publications shows each title's cover, with no text labels for the icons. Each selection is purely graphical, which causes TalkBack to keep repeating "Column 1." For this reason, it is impossible to select the desired title. Only after the publication has been opened is its title spoken. There is therefore no way of choosing which magazine to read without the user having to tap to open each one until the desired title is found by chance. It is possible to limit the results in this list by performing a search (by sorting by category, country, province and language,) but text labels are still crucial in this list.

Recommendation. Have all list items clearly labeled in search results.

The list of items is designed to be scrolled through with a single-finger swipe to the right. TalkBack utilizes this gesture to move between controls, however a three-fingered swipe up in the list does work when TalkBack is enabled. The problem is that the app doesn't indicate how many publications are available in the list, or even that the list can be scrolled.

Recommendation. Include an audible indication that more publications are available; a simple "Swipe for more" would avoid the situation where the user is unaware that more titles are available.

Applying these recommendations will make searches less confusing and easier to navigate.

Downloading Titles

iOS with VoiceOver

Each item in the list of results is associated with a button labelled "iArrow," which does not describe the button's function to download content. Once an article is available, the button changes to something labeled "iReady," which also does not clearly describe its functionality.

Recommendation. Label the button so that the label and the button are all on one element; use a meaningful label, such as "open" or "read."

When the user downloads a publication, the screen begins refreshing once every few seconds. This severely impairs the ability to navigate the app at all, as every refresh causes VoiceOver to stop speaking, and sometimes to lose focus when the screen changes. This happens when PressReader is downloading a publication, and the screen refreshes happen all throughout the app so it becomes very frustrating to use until all downloads have finished. For instance, if the user downloads a magazine, they can tell that the screen refreshes every few seconds, but it is not possible to know what it is doing (possibly auto flipping through pages.) When the user stops it from refreshing, the button then turns into the "iReady" button.

Recommendation. Allow background downloads, so that users can continue using the app while a download is in progress.

To remove downloaded magazines/newspapers, the user must go to the "iCustomize" button and click on edit, but the edit screen presents the same problem where each item has an unlabeled button associated with it. When tapping the "Add to My Publication" button, it changes to simply read "My Publication." For clarity, it should be something like

"Unfollow" or "Remove from My Publication." Pressing this button a second time does in fact remove it from My Publication and changes it back to an add button.

Recommendation. Label this button so that the label and the button are all on one element, and use meaningful labels.

By applying these recommendations it will lessen user frustration when downloading a title from the app and make it clear when a title is ready to be read.

Android with TalkBack

Next to the cover of each publication is a "Download" button. However, this control lacks not only a text label, but also the designation of a clickable element. For these reasons, TalkBack says nothing when this button is selected.

Recommendation. Make sure the controls for downloading are clearly labeled and indicate they are a clickable element.

By applying this recommendation it will be easier for users to download titles from the app.

Reading and Navigating Content

The reading and navigation content user experience is different for iOS and Android. The testers for iOS found that accessing and reading magazines and newspapers was similar, whereas the testers for Android found them to be different. For this reason the test results and recommendations section for Android has been broken down into separate sections for magazines and newspapers.

iOS with VoiceOver

There is an unlabeled button associated with each magazine/newspaper on the downloaded tab when it is set on grid view. When the user clicks on the "iCustomize" button to change the view from grid view to list view, the unlabeled button containing the name of the newspaper/magazine is replaced with an "AM Radio" button, which reads the article using text-to-speech.

Recommendation. Label the button so that the label and button are all on one element, and use meaningful labels. Also, consider removing the letter "i" to make the names more descriptive.

The default reader is page view which is not accessible as it is image-based. In order to read articles, a user relying on VoiceOver has to go to Customize and change the view to text. As it is currently setup this switch is difficult to access, and is not an intuitive process. The users cannot quickly navigate to different magazines/newspapers, which detracts from the reading experience. For readers relying on assistive technology, text view is the optimal option, and it should be easy for the user to switch to this view without any barriers. Currently VoiceOver users need to tap-and-hold the "iCustomize" button to activate it, then get past some unlabeled checkboxes to the text view option.

In the customize screen, there are also some buttons that have unclear labels that read as “iCheckboxRisedUnchecked.” These can be found when swiping past the heading that says “1 page selected.” Many of the buttons are grayed out, which means that they are not clickable; they are identified by VoiceOver as “dimmed” and users cannot interact with them.

When the user clicks on the text view, they are taken to a screen from which they can navigate to different sections of the newspaper/magazine. Note that the “Back,” “Talk” and “Customize” buttons are labelled with an “i” at the beginning of the label as well. While the options button labelled as “IC Options Small” is labelled oddly, it is possible to interact with it and understand the relationship between it and the article it is referring to, because it logically makes sense in the swipe order. Nevertheless, even if there is logic to the swipe order, all buttons should have clearer labels.

Recommendations. Have the app default to text view when VoiceOver is active, or have a dialog come up asking VoiceOver users if they want to switch to text view for improved accessibility. If neither of those options is possible, the button to customize the view should be easier to activate, or it can be an option in the settings section. Ensure all buttons are clearly and purposefully labeled, and are clickable by screen readers.

When in text view, each article is presented as one gigantic block of text that is contained in one element. This makes it difficult to use VoiceOver gestures to explore the text granularly. Note that the “Back” button here is labelled as “icBackRised.” There is also an element that has a single letter when the user swipes past the article, which does not make sense to users relying on VoiceOver.

Recommendations. Consider breaking articles into paragraphs so the “Read from Current Position” feature works more reliably in text view. Make sure that each line of text has its own element and that a heading is added, so that users relying on a screen reader can quickly jump to the beginning of the article using VoiceOver commands, and navigate the text easily.

When the user relying on VoiceOver attempts to read magazines/newspapers, they sometimes encounter a problem where text and controls disappear. For example, after using the double-tap gesture to open a magazine (a gesture specific to VoiceOver to simulate a single tap on an item,) VoiceOver reads the “Back” button, the current page the user is on, as well as the “Talk” and “Customize” buttons. This only lasts a moment before the “Back” and “Talk” buttons, along with the information displaying the current page, disappear very quickly. The only thing VoiceOver reads at this point is a “Customize” button. This means that all those other elements on the screen are invisible and there is no way to navigate to those elements. This makes it very difficult to move back from this screen, and the only workaround is to click on the “Customize” button, open text view and then click on the “Back” button to go to the previous screen. It is only then the “Back” button and other elements are visible again for a short time. From here the user has to quickly click the “Back” button once more to go to the downloaded section before those elements disappear. While it is possible to read articles, this inconsistency of the navigational features makes it difficult to do so. Note that the "Back," "Talk" and "Customize" buttons are labelled with an "i" at the beginning as well.

Recommendation. Ensure all navigation elements are accessible in Text View at all times.

By applying these recommendations, navigating and reading the resources provided by your app will become more accessible to readers with visual impairments.

Android with TalkBack

As mentioned earlier, Android testers found that the reading and navigational experience for magazines and newspapers different, although the testers did note a few similar problems for both resources. They found clicking on the name of a newspaper or magazine will bring up another screen with the list of issues. However, the user can't tell which issue is selected because after navigating to "Latest" (assuming it means the latest issue,) they encounter several blank elements and unlabeled buttons. The user does not know what those buttons are, and they cannot select issues easily by date. If a user clicks on an individual issue, such as the latest issue, they can find a few options, but these quickly disappear and the only option left is "Switch to Text View." Even when the user changes the view to text, they are unable to see any of the text of articles, and again they encounter a number of unlabeled buttons/blank elements. The only thing that the user relying on a screen reader can do here is see the table of contents and switch to different sections.

Below you will find a more detailed breakdown of the test results and recommendations for the two different resource types as accessed through Android.

Magazines

Once a magazine opens, its title is spoken and a "Listen" button appears near the top of the screen, which seems to simplify the interface considerably when clicked. Publication, and other unrelated information disappears, and is replaced by controls to move through the magazine to start reading it. Though the "Listen" button is presented to TalkBack, it doesn't seem to actually be displayed visually.

Recommendation. Please ensure that all controls that can be seen on the screen are read by the screen reader. Controls that are only accessible to a single audience make it difficult for one group to offer assistance to the other.

After clicking "Listen" the app did start processing and opened the magazine, but there was no audible indication that the request to play the content was in process. By swiping around the screen the tester was able to find the title of the first article, but also discovered that the title is followed by seven unlabeled buttons. The tester was able to determine that the fourth button is the "Play" button, but before the user could select this button it started to play on its own. This did not cause an issue until the tester wanted to navigate to a different section, or pause the audio. The magazine would continue to play, and the playback volume is at the same level as TalkBack, which makes performing additional navigation after it starts to play rather challenging.

Recommendation. Clearly label all buttons for ease of functionality. Also, ensure that the voice for assistive devices like TalkBack are louder than the playback of the item, or have the playback volume lower when TalkBack is being used for navigation.

Testers also found that navigating the magazine is not simple. Once it starts playing, the user must swipe down to reach the article name. A three-finger swipe will move forward and back through the articles, which is helpful for users. However, once the magazine starts reading the user must swipe back through the unlabeled buttons with a single finger, counting them until the "Pause" button is reached.

Recommendation. Simplify the reading interface for ease of use, such as having the "Listen" button open an accessible view with large buttons for "Pause," "Back" and "Next."

If the app is closed, without pausing first, the magazine continues to play even after pressing the "Home" button to return to the device's home screen. As mentioned above, the synthetic voice reading the magazine is louder than the voice for TalkBack, which makes the app unusable at this point. Sighted assistance is required to reopen the app, select the appropriate magazine, and press the "Pause" button. It is critical that the magazine stops playing after it is closed, and when a user navigates to another page of the app.

Recommendation. Ensure the reader stops reading when the user navigates to a new page while using a screen reader. Ensure the "Pause" button is clearly labeled and easy to find.

Text labels are missing from the button-bar for "Like," "Dislike," "Back," "Play/Pause," "Forward," "Mute," and "Volume Up." This makes it impossible to enjoy these features without pressing each button at random to find out what it does. Graphical controls without text labels give no audible indication of the control's purpose, so their function remains a mystery when using a screen reader.

Recommendation. Ensure all buttons and navigation bars are clearly and appropriately labeled.

It is worth noting that the "Mute" button poses a significant hazard to users who rely on audible feedback to use the app—and even the user's device itself. One tester reported that after accidentally clicking the "Mute" button their device was rendered inoperative until sighted assistance was available to unmute the audio.

Recommendation. Consider creating pop up confirmation of "Yes" or "No" before muting the device to ensure that users of screen readers do not accidentally mute the reader and their device. Also consider have the "Mute" button not as prominently displayed.

Depending on the speed of the user's connection, there can be a significant delay between when the "Play" button is pressed and when the synthetic speech actually starts reading the article. This wouldn't be an issue if the user knew the app was working. As it is, however, it gives the illusion that the request to play a section was ignored, appearing to do nothing until the speech suddenly starts playing. Testers also noted that the “Back” and “Next” buttons move to different headings within the current article, but there is a significant delay before the new subsection starts to play. Visually, a swirling circle indicates that the audio is loading, but no information of this fact is picked up on by TalkBack.

Recommendation. Add a simple audible notice of “Downloading, Please Wait” after the “Play” button is selected. Build streaming into the delivery mechanism so the audio could start nearly instantaneously, or feeding the app text so the user could read it with the screen reader.

It was also noted that the listen feature does work to read a publication, however the text is completely ignored by TalkBack. Testers found that TalkBack will read the title of the current article, but nothing more. For this reason, the body of an article cannot be reviewed, copied, or spelled.

Recommendation. Ensure that the text of the articles is accessible to TalkBack; consider showing articles in text by default when the screen reader is active.

Newspapers

The newspaper view is vastly less accessible than the magazine view. The screen opens on a full-page graphical experience, which hides even the Android’s “Back,” “Home” and “Overview” buttons at the bottom of the screen. In order to pull up the navigation pane the user must first double-tap anywhere on the screen.

Recommendation. Consider defaulting to text-view to show articles when the screen reader is active.

Testers also noted that access to certain buttons is only available for brief moments. When reading, a user must quickly swipe over to and click the “page_slider” button before the controls disappear, and then the user is returned to the graphical view. This causes an inconsistent and frustrating interaction since the navigation menu is so prone to appearing and disappearing. Users with print disabilities may have difficulty using a mouse or keyboard, and rely on assistive technology, which means they require more time to complete tasks.

Recommendation. The controls and navigation menus should be static on all pages and not disappear.

Once “page_slider” is pressed, it is possible to swipe over to the list of pages. Though the titles of these pages don’t usually read, it is still possible to swipe down to a desired page and click it. In addition to unlabeled buttons for various pages, which were discovered to be “Share,” “Listen” and “Save to Collection,” there is an icon of three round circles at the top right-hand part of the screen. However, this icon is so small, and so close to the edge of the screen that TalkBack doesn’t recognize it as a clickable button. It was also noted that in order to select the “Listen” feature, TalkBack must be temporarily disabled, which is counter-intuitive to the use of assistive technology.

Recommendation. All buttons should be large, prominently positioned, and labeled clearly and appropriately.

Clicking the “Listen” button brings up the unlabeled six-button block reminiscent of the magazine view. Again, certain controls are only detectable by swiping a given way, and the controls will not loop around so they can be found by swiping in a single direction. This makes it challenging not only to find all the controls that are available, but to get a good mental picture of how the screen is laid out.

Recommendation. Design and label the buttons in an order that is logical and easy to navigate with swiping gestures.

Visual Adjustments

The options for visual adjustments in the mobile app are not aimed at people with low vision. More options are needed to allow people to easily change font type and size, colour contrast, and text spacing to meet their specific needs.

iOS

Other than iOS-provided magnifying glasses, the app does not seem to have a built-in magnification option for its user interface. When reading page-by-page it is possible to enlarge an individual page, which makes the text much larger. Spreading two fingers apart magnifies a certain part of the page to make the text even larger in one place. This feature can be turned off if desired.

When the menu is pulled up, pressing the button one over from the bottom left pulls up a list of articles in the magazine. The icon of three circles vertically placed on top of each other at the top left of the screen pulls up a “Fonts” button, which lets the user select Small (about 8 pt.) or Large (about 10 pt.) font. Users can also select the font face when changing the font size, but the difference is too small to help low-vision readers much. Text is even smaller in print-heavy newspapers, leading to more accessibility issues for readers with low vision.

Recommendations. Add a function to customize the color of the font or its background. Allow changing brightness, add pre-defined style themes, and the option to adjust text spacing and alignment or margins.

Adding these options will help users with low vision easily change the text to meet their specific needs. People with low vision have a wide variety of issues when it comes to accessibility, so giving the ability to customize the text and background will ensure optimal reading access.

Android

The in-app settings for visual adjustments were designed more for people with mild vision issues; not for people with perceptual disabilities. While it is possible to use a device’s accessibility settings for magnification, they work better with the settings within PressReader turned off.

Recommendation. Allow for a wider range of visual adjustments, such as changing the background and font colour, spaces between words and letters, and being able to highlight text in multiple colours.

As mentioned above with the iOS devices, it is best to give a wide range of options to the user with low vision needs to ensure the app is fully accessible to those who actually have visual impairments

The PressReader Website

Overall, testers using Windows, Mac OS and Chrome OS reported their experience as frustrating; they all concluded in that they would not recommend the PressReader website for screen reader users.

In this section of the report, we highlight the problematic issues as they relate to the different parts of the PressReader website as it was tested in Windows, Safari, and Chrome browsers; but we do not elaborate on each of the features. To see the entire list of questions, please refer to Appendix A.

Since many of the findings related to the websites are similar across the different sections, we describe these issues more broadly, and present specific examples to illustrate some of the barriers relevant to each operating system.

Windows Browser: with JAWS or NVDA

The accessibility priorities as identified by our testers for Windows browsers include:

- Labeling buttons and controls correctly;
- remove hidden text/invisible server error messages;
- simplifying the website structure.

The testers found that the structure of the website makes it difficult to find things. Many of the pages are text heavy, and when it is presented audibly the information doesn't have a clear organization. Our testers found that several pages have long lists of items that are all styled with Heading Level 1, and often with labels that do not make sense. For instance, at the bottom of many pages there seemed to be options to switch language, but they appeared as a list of over twenty Heading Level 1 items that are all read as “English.” The lists of articles also appear with all their titles at Heading Level 1, which makes navigation difficult for people using screen readers to skip past them to other sections of the site. Similarly, in the advanced search dialog when the user presses enter on a link called “Publication” it expands to a list of hundreds of countries, each with a checkbox next to it; however, it is not easy to know which item each checkbox corresponds to.

Recommendations. Use headings in logical cascading order to improve the ability of screen reader users to navigate the website. Ensure all labels are appropriately and functionally marked. Consider using lists to improve navigation for screen readers.

The testers found that the custom controls (such as grids of displayable buttons, or clickable items of a nonstandard type) were difficult and cumbersome to use with the screen readers. For example, to navigate the searches all the controls and list items are graphical. They also found that the open button is unlabeled, and it is also not possible to access it by using the tab key. This means that in order to activate the reading function a user must find and click two unlabeled, untabbable, nonstandard controls. This can only be accomplished with trial and error causing high user frustration in the process. The options for filtering searches are also not tabbable or properly labeled, so users relying on screen readers cannot filter searches.

The buttons for Magazines and Newspapers are also unlabeled; a list of items that follows are generically called "NewsPaperVmItem" that are actually different newspaper titles with the same improper label. The testers also found that to read a given issue the user has to click on "Publications," which NVDA indicates is the fourth item in the main list. Unfortunately, this was labeled as a "MainMenu VM item" instead of a correct publications label, which lead to more confusion for the user. Standardization of controls are always better for accessibility as it makes navigation and identifying different controls of the site easier to understand.

Recommendations. Use standardization for controls instead of customization, and ensure the controls are tabbable. Ensure that buttons and items are labeled correctly.

The testers also reported that there are too many controls to efficiently navigate the website. This also refers back to the site being too text heavy. The site itself has too much information on it that is not organized in a way that can be easily accessed using a screen reader. Again, cleaning up the navigation and controls would help to fix this issue.

Recommendation. Consider utilizing a standard pulldown menu to clean up main screens, or at the very least enclosing like-minded controls within tabs or group boxes.

The testers found that the lists are also unlabeled. The list that contains "Home Feed," "Publications" and "Downloaded" reads with NVDA as "Profile and Main menu.vm" items, and many of the links in this list are disabled. Moreover, this list of items was not identified as clickable by the screen readers, so the testers were unsure what the list was for and had to figure it out through trial and error. Another example is the subsequent list, containing the "Downloaded" icon, which reads as "My Library." Having icons voiced differently from the way they appear on the screen is not only confusing for those using the platform with a screen reader, it inhibits their ability to converse with sighted users about their experiences and obtain assistance. There is also a "Sound" button, which reads as a long string of alpha-numerics headed with "M14-20.5..."

Recommendation. Caption lists with a proper and descriptive label of their own. Ensure that items are identified as clickable and functional.

With regards to reading the content, most of the text is broken up with ascii 173 characters that look like hyphens in the middle of words. This means that a screen reader will pronounce these words incorrectly. In order to read an article it would be necessary to cut and paste the text into a Word processor and do a search-and-replace to remove the hyphens. Testers also found that the "Listen" button did not open for all publications, and was almost impossible to locate for both magazines and newspapers.

Recommendations. Ensure that the text is coded with no "ascii 173 characters." Ensure that the buttons to access the reading features are easy to locate and functional.

Testers across Windows browsers found that layout and navigation design was too complicated with too many controls to efficiently navigate the app.

Recommendation. Utilize a standard pulldown menu to clean up main screens, or consider enclosing like-minded controls within tabs or group boxes.

Testers also found that the log-in procedure was confusing. Although the process to sign in worked well with username and password, finding the sign-in button was quite difficult. Again, this button was buried in will all the other controls. There was also no audible indication that sign-in had been successful, or that it persisted with subsequent sessions of the app. As one tester pointed out:

> "Spoken announcements of important information would be less necessary if the layout of the site were improved to make finding information easier. Currently though, I spent a lot of time just finding out if I was logged in or not." (NNELS accessibility tester, February 2019.)

Recommendations. Links for managing accounts need to be labelled in text to make them accessible to screen reader users. The Title Bar - next to the name of the App - is a great place to show signed-in status. Another alternative is to have spoken alerts, via ARIA, to let users know when they logged in successfully and when they are not.

Another issue was that the screen readers in Windows read text that is not displayed visually on the screen. In particular, testers kept getting an inaccurate server error message, which was read at different points during their testing; the error reads: "500 - Internal server error. There is a problem with the resource you are looking for, and it cannot be displayed." Again, this error is not visually present, but the screen readers kept picking it up as the testers explored the app. This created unnecessary confusion.

Recommendation. Ensure that no invisible server errors are present on the website, to prevent screen readers from picking this information and creating confusion.

Applying these recommendations will improve navigation and readability of the website for screen readers.

Mac OS Safari with VoiceOver

Although the PressReader website can be used in Safari with VoiceOver in Mac OS, VoiceOver stumbles over certain parts, and the behaviour of some sections of the website is not consistent, which make some of the core functionality difficult or impossible to interact with.

The accessibility priorities as identified by our testers for Safari browsers include:

- making article names into links;
- streamlining text view so that individual articles are rendered properly; and
- determining the reason for certain links not working at all.

The main issue identified by our testers is that there are many links which, if clicked, appear to do nothing. For example, once you open the link to the navigation menu it stops working. This makes it impossible to close the menu, causing it to stay on-screen until the user refreshes the page. Also, within the menu, the settings link did not work, making it impossible to change settings. The testers also discovered that the "Continue Reading" link on articles also did not work; this prevents screen reader users from reading articles in full.

Recommendation. Ensure all links are active throughout the website.

When it comes to selecting an article to read the testers found that none of the title links worked with their screen readers. The keystroke to activate a link or button (Control+Option+Space) doesn’t work when the user clicks on the name of a title; this is both in the search results and in any of the publication lists from the home screen, as well as on the publications page. This creates a major problem with VoiceOver because users are not able to use the typical keystroke to activate these. The testers did figure out that the article names are clickable pieces of text through trial and error. VoiceOver does not recognize them as clickable, so users have to use the VoiceOver command to route the mouse and click on a title to open it. This is not a reliable solution, since it only proved to work some of the time and was time consuming to figure out how to troubleshoot.

Recommendation. Make all article titles clickable links that can be easily activated using a keyboard.

For the search results, there is no defining start to the results, and they are pieces of text with no element type, or ARIA role, to make them navigable by a screen reader. A "Search Results" heading above the first page would help orient the reader.

Recommendation. Results should be clickable links with headings, and the results should be part of a navigable list.

It should also be noted that Safari users of Mac OS using VoiceOver could not locate the “Advanced Search” button, though this function was accessible through Chrome users on Windows with NVDA or JAWS.

When it came to reading an article, the testers found that the page view is not accessible. They were able to switch to text view, but found that moving to different sections in a publication is inconsistent and confusing. Text view shows the contents, but clicking a link to a given section does not actually move the screen reader's cursor to that section. Instead, the entire publication appears separated by headings. This is not a huge problem, but since it is possible for links to move the user to a certain spot on the page, it would be good if screen readers could also do this successfully. Users are also not always able to reliably click on the link to read a full article, and, as mentioned earlier, sometimes the links do not work. Testers also found that the "Continue" link did not seem to work.

Recommendation. Ensure that navigation of the article through links is compatible with screen readers.

Fixing these issues would bring the website in line with good accessibility practices, and would make it infinitely easier for all VoiceOver users to enjoy.

Chrome OS with ChromeVox Screen Reader

An important accessibility barrier is the dynamic content changes on the page. When the user clicks on different parts of the page such as any of the "Continue" links, there is an option to select the magazine that appears at the bottom of the page. Another example is that when a user clicks on any other articles in a magazine, the "Next Text" and reading controls appear near the top of the page. This can be problematic because users may have a difficult time knowing whether any new content has appeared, or even be able to locate where the new content is.

Recommendation. Send an alert to let the reader know that the new article or options are available. Also, it would be helpful to always put new content at the bottom and add a heading to indicate the beginning of the section. An alternative is to move the focus to the beginning of the article, or to any new options that appear.

Similar to how the website renders in Windows, in Chrome OS many of the links are unlabeled, including the links prior to "Previous," "Play" and "Next." Troubling shooting this issue did not yield any changes that the tester could make sense of, and they were unsure why these links were marked. The homepage also has several ambiguous links that were labelled as "Continue" or "See More," which caused confusion since they are presented with no context.

Recommendation. Use meaningful text labels for links.

Some of the clickable elements were not indicated as being clickable with ChromeVox. For example, to select a magazine the user has to press enter on the title, but since this simply appears as a piece of text without any indication of its role, some people may not know the titles are clickable. Using tabbing and shift to try and troubleshoot this was inconsistent, and does not work in the menus. The tester was unable to tab or shift-tab to move through menu items at the top of each page.

Recommendation. Ensure clickable elements are marked as such so users can efficiently navigate through the pages. Perhaps changing the elements to buttons as opposed to linked text would help solve this issue.

The testers found that when they opened a magazine, they could find two buttons that change the item from page view to text view near the top of the page, but these options are repeated again as links further down near the bottom of the same page. Having these two options causes confusion for the user. This is another example of there being too much information on the page. When it came to selecting text view, the testers did not notice much change in the item, and could only read snippets of the whole even though it was supposedly showing the entire item. The tester was unable to find a way to expand the snippet of text, but interestingly, when they clicked on "Listen" they could read more of the article. The tester also found that when reading a magazine there are several lines with "aA" at the top of the page. The tester was not able to determine if these were supposed to be clickable elements or not, and noted that there are also several empty headings with an article element tied to them that can be found just past the content links. It was not clear what they were for and caused confusion when trying to read the item.

To help make reading easier it would be good to have more playback controls such as "Volume," "Increase/Decrease" playback speed, and "Jump by X Number of Seconds" as part of that page along with keyboard shortcuts that do not interfere with screen reader commands. Adding Alt+Shift followed by a letter such as "p" for Play/Pause or Alt followed by a letter would be a good design choice.

Recommendation. Create more playback controls that are easy to find and use for reading with a screen reader.

Fixing these issues would bring the website into line with good accessibility practices, and would make it easier for all ChromeVox users to enjoy.

Conclusion

In conclusion, the PressReader application for iOS and Android, as well as the website, have several accessibility issues that our testers identified as barriers for the average user. Testers could not easily navigate the mobile apps and the website due to all the unlabeled controls and buttons, as well as the complex interfaces. This means that users relying on assistive technology cannot fully enjoy the impressive collection of resources that PressReader offers. The recommendations in this report are aimed to help with improvements that would make this app a viable option for users who have print disabilities.

Contributors

The following people contributed to this report, including testers and editors.

- Karoline Bourdeau
- Danny Faris
- Kaden Faris
- Simon Jaeger
- Daniella Levy-Pinto
- Ka Li
- Richard Marion
- Steve Murgaski
- Rachel Osolen

Appendix A: Evaluation Questions

Testers answered the following questions using an Excel document to record their findings and provide relevant details. The first questions are about the application tested, the platform and screen reader used. The rest of the questions require a yes/no answer, or a 1 to 5 rating of the accessibility of each function: 1 (not accessible) to 5 (very accessible.) If any of the questions was not relevant to the application tested (e.g. if there are no options for creating notes) testers recorded N.A. (not applicable.)

- Application name/version
- Operating system
- Did you use a computer or a mobile device?
- Which screen reader did you use?
- [1-not accessible/5-totally accessible] How accessible is the process to sign-up?
- [1/5] How accessible is the function to login with library barcode/pin?
- [Yes/No] Is it possible to use email to sign up/login?
- [1/5] How accessible is the function to set preferences.?
- [1/5] How accessible is the function to search for articles?
- [1/5] How accessible is the function to navigate search results?
- [1/5] How accessible is the function to check out articles?
- [1/5] How accessible is the function to explore bookshelf/my account for titles checked out, usage history?
- [1/5] How accessible is the function to delete an article?
- [1/5] How accessible is the function to navigate and use curated lists?
- [1/5] How accessible is the function to limit/filter searches?
- [1/5] How accessible is the function to navigate help screens?
- [1/5] How accessible is the function to log out of the application?
- [1/5] How accessible is all the reference information about articles (publisher, year, number of pages etc.)?
- [1/5] How accessible is it to activate the reading function?
- [1/5] How accessible is the function to access "where am I" information (percentage, time, page number)?

  - [Yes/No] Is it possible to read from the currently focused position in an article, rather than beginning at the top?
  - [Yes/No] Can you pause and resume reading at the same location rather than changing focus to the top of a chapter/section?
  - [Yes/No Is there a "read-to-end" function?
  - [Yes/No] Can you adjust the reading speed from within the application?
  - [Yes/No] As you read an article, does the TTS or your screen reader pause to indicate the beginning of a new heading or section?
  - [Yes/No] Can you read skippable structures like sidebars, annotations and producer notes?
- [Yes/No] Does the app read image alt text?
- [1/5] How accessible is the function to move from the index/table of contents to different sections/articles within a publication?
  - [Yes/No] Can you navigate an article by headings separating sections (e.g. chapters)?
  - [Yes/No] Can you navigate the article by page?
  - [Yes/No] Can you navigate by reflowed page number?
  - [Yes/No] Can you move by character?
  - [Yes/No] Can you move by word?
  - [Yes/No] Can you move by line?
  - [Yes/No] Can you navigate within a table (e.g. move to next cell or row)?
  - [1/5] How accessible is the function to navigate within internal hyperlinks?
  - [1/5] How accessible is the function to search for a string of characters and move to the results?
  - [1/5] How accessible is the function to add bookmarks or highlights?
  - [1/5] How accessible is the function to add notes?
  - [1/5] How accessible is the function to review bookmarks and notes?
  - [1/5] How accessible is the function to view a single note?
  - [Yes/No] Does the application support Braille devices?
  - [Yes/No] Is it possible to export content to a different application?
  - [Yes/No] Is it possible to copy text content in the application to the clipboard?
  - [Yes/No] Can the font size be changed?
  - [Yes/No] Can font colour and foreground colour be changed?
  - [Yes/No] Can font type be changed?
  - [Yes/No] Can brightness be changed?
  - [Yes/No] Can highlight text color be changed?
  - [Yes/No] Can text, word, and line spacing be changed?
  - [Yes/No] Can alignment (left, center, right) be changed?
  - [Yes/No] Can margins be changed?
  - [Yes/No] Can user visual preferences (style sheets) be changed?
  - [Yes/No] Can all styling be removed?
  - [Yes/No] Is it possible to select a pre-defined style theme or honor high contrast system configuration?
  - [Yes/No] Is it possible to change from paginated to scrolled view

- [Yes/No] Is it possible to change from single to dual page view
- [Yes/No] Does the application support UI magnification?
- [Yes/No] Do all buttons/controls have text labels? Please provide details.
- [Yes/No] Does the links' text adequately describe what the links do? Please provide details.
- [1/5] Rate your overall experience using this application in a 1/5 scale (1 very poor and 5 excellent)
- List the three top priorities to enhance accessibility of this application.