

NNELS



Accessibility Report for Rocket Languages

July 2022

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](https://creativecommons.org/licenses/by-nc/4.0/)

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 disabilities¹. 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

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

print disabilities, including blindness, low vision, and learning disabilities.

Introduction

Rocket Languages is an online platform which provides access to language learning resources through participating public libraries. The service is offered as a mobile application and website. NNELS was provided with test accounts through the Northwest Territories Public Libraries.

Our testers used screen-reading and magnification software to assess the usability of the apps and website across supported platforms. Readers can find a complete list of all the software and operating systems used in this assessment in this report's *Systems and Assistive Technology* section.

This assessment aims 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 Rocket Languages itself. As a result, some functionality may not be discussed at all or in-depth.

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 the 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 colour contrast, or a combination of some or all 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 Summary

Rocket Languages contains significant accessibility barriers, preventing users with print disabilities from having an equal experience. On all tested platforms, Rocket Languages suffers from poor heading hierarchy and page structure, inconsistent or missing button labels, and other markup issues, leading to difficulty navigating the interface with a screen reader. On Android, most controls have a missing control type, a missing label, or both. On iOS, screen reader users could not start language lessons. The only way to play the lesson audio is through the playlist view. Additionally, when using flashcards on the web, the answer is visually hidden until a reveal button is

pressed, but it is readable with a screen reader before the button is pushed.

Low-vision testers found the interface was uncluttered and often easy to use with magnification, but the inconsistent colour schemes and lack of adjustable font impaired readability. Additional visual adjustments—such as font, text size, and ability to change colour schemes—would go a long way toward improving the experience.

Accessibility Performance and Recommendations

This section will dive deeper into specific accessibility issues encountered while testing the Rocket Languages apps and website. Below you will find the testing results and their related recommendations as they pertain to:

- Library Access
- Signing In
- Language Learning

Finally, the *Development Recommendations* sections contain suggestions for improving the interface on each platform. These suggestions will be relevant to any issues or observations noted above.

Rocket Languages Website

Signing In

Rocket languages require users to create an account in addition to signing in with a library card. On the account creation screen, none of the text fields have labels assigned to them. A screen reader user can explore the page above each text field to find the label, but the label is not associated with the text field. It is, therefore, not announced automatically, making this process unnecessarily cumbersome.

General

The homepage has a menu button that does not appear to be accessible with screen readers. Testers reported that pressing it had no effect except for the addition of a "Close Menu" button.

The heading hierarchy on the homepage is incorrect: The list of courses is under a level-3 heading, which should only be the case if there is a level-2 heading containing it. There are no level-2 headings on the page, and two at level 1, which is considered bad practice since level-1 headings typically denote the start of the page's main content.

Similarly, on the help page, each question is under a level-3 heading with no level-2 before it. The page also does not indicate whether each question's answer is expanded or collapsed. While it is possible to read past the question and check for an answer, screen reader users could skip this step if the expanded and collapsed states were correctly reported.

Language Lessons

After selecting a language course, a list of modules is presented. Each module contains a list of lessons which can be expanded (shown) or collapsed (hidden.) Screen readers correctly indicate whether a module is expanded or collapsed.

Lessons are shown as clickable controls rather than buttons or links. The lack of control type is difficult for screen reader users who may want to move quickly between lessons.

Each lesson can be expanded, revealing information about the lesson and a "Start" or "Continue" link (depending on which is appropriate). Screen readers do not indicate whether a lesson is expanded or collapsed.

A short tutorial is available by clicking the "Find out how everything works!" link. This tutorial contains several unlabeled buttons. After completing or skipping it, the keyboard focus lands on a random lesson rather than returning to the "Find out how everything works!" link.

Once a lesson has been started, the interface has several accessibility barriers: the "Flashcards," "Hear It Say It," "Write It," "Know It," and "Quiz" buttons are unlabeled and do not have a control type. These should ideally be buttons or links and should have descriptive labels. Testers were only able to determine their function by pressing them.

The lesson's name is repeated twice under two separate level-1 headings, whereas the lesson number is a level-4 heading. This was likely done for stylistic reasons, but the resulting heading structure is incorrect.

Each lesson can be listened to via a set of play controls on the page. These are located under a "Listen to the Full Lesson" heading. The seek slider is a custom control inaccessible for keyboard and screen reader users. The audio can be downloaded as an MP3 file, so users impacted by the inaccessible slider can play the files elsewhere as a temporary workaround.

The "Play It" section lets the user play the role of one of the characters in the lesson. Screen reader users noted that errors were not announced, leaving the user to check for them manually.

"Flashcards" display a word or phrase on one side and an answer on the other. The website simulates this behaviour with a "Reveal" button which causes the answer to be shown. For screen reader users, the answer is always shown regardless of whether the button has been pressed. To create an equal experience, the screen reader should not detect the answer until it is visually displayed.

The "Hear It, Say It" section has shortcut keys to play a phrase and start recording. This is a useful accessibility aid since it eliminates the need to simultaneously listen to a screen reader and a spoken phrase.

The "Write It" section has extraneous unlabeled buttons before the "Reveal" button and each difficulty rating. These buttons appear to perform the same function as the control following

them, but the extra controls create more page clutter and potential confusion for a screen reader.

When searching for a phrase within a course, screen readers do not indicate when the results have loaded. This can be easily accomplished using an invisible piece of text with an [alert role](#). Search results are presented as level-3 headings but fall directly below the "Search Results" heading at level 1, so these results should instead be level-2 headings. Lastly, the search field has an unlabeled image below it.

When accessing the settings for each course, a screen reader user cannot determine which difficulty level is selected. Semantically, these difficulty-level selectors should be radio buttons.

Vocabulary

The vocabulary tool allows for saving specific words to the account. The list of words is presented as a table, but screen reader navigation commands (such as alt+ctrl+arrows in NVDA) do not function in this table. Unfortunately, the mechanism for saving words is not accessible because it requires the use of a mouse. For this feature to work for keyboard and screen reader users, the controls should be changed to buttons or another type of interactive element.

Creating Flashcards

Users can create and share custom decks of flashcards. This tool has several major accessibility concerns.

When creating a deck, the edit fields are unlabelled, so a screen reader will not announce the text field's name, and users may not know which information to enter.

When browsing through decks created by other users, each deck is presented in a table. Clicking (or otherwise activating) the name of a deck will present more information about it. Similarly to the list of lessons, screen readers will not announce when these decks have expanded and will not indicate which ones are

expanded or collapsed. The only way to determine this is to read through the page and look for the extra information. The same issue can be observed when pressing the “Settings” button after expanding a deck. These controls would benefit from using the [aria-expanded](#) property, which would cause screen readers to announce the new expanded or collapsed state.

Visual Appearance

Although our specific team of testers found the website visually intuitive and easy to use, there are no Visual adjustments. The website would benefit from an adjustable font, spacing, and colour scheme—particularly the option to switch to a less complex one.

Development Recommendations

- Improve heading hierarchy across the site.
- Allow for adjustment of visual appearance, particularly the font and colour scheme.
- Within a module, change lessons to links or buttons rather than clickable pieces of text with no semantic markup.
- Ensure that lessons are correctly reported as “expanded” or “collapsed” when clicked.
- Label all controls within lessons.
- Do not reveal the answer side of a flashcard to screen reader users until it is revealed visually.
- Allow selection of vocabulary words with a keyboard (e.g., by making each word a button.)

Rocket Languages App for iOS

- Tested iOS version: 14.6+
- Tested app version: 5.6.5

Signing In

When using the app with VoiceOver, the labels on the login screen are each read twice—once before the text field and once after it.

General

The “Sidebar” and “Options” controls do not have a control type, so VoiceOver users may not know they are clickable. For clarity, these should be changed to buttons. There is a blank and seemingly redundant control after the “Options” button, which activates the sidebar.

When the “Options” control is pressed, a menu appears at the bottom of the screen. VoiceOver users must explore the screen or jump to the last element to find this menu. Instead, consider using a popup menu or automatically setting VoiceOver focus to the first option when the menu opens.

On the stats and leaderboard screens, the main table lacks some necessary markup which would allow a VoiceOver user to navigate through the rows and columns easily. VoiceOver’s rotor allows one to swipe up or down to move by row, which does not work here. Making sense of the information is more difficult without the ability to navigate the table spatially. The back button on the stats screen is also labelled “Level 1.”

The leaderboard allows filtering by time or course. However, the controls for applying these filters do not have a role or label, so VoiceOver reads nothing when navigating to them.

Once a filter has been opened, the “Back” button is a blank and nameless control, and the “Done” control has no button role. VoiceOver users also found that the “Up Arrow” and “Down Arrow” buttons appear to do nothing when pressed.

The “Profiles” screen (accessed from the options menu) has the same text field problem as the login screen: Each label is duplicated before and after the text field. Other controls (such as “Update Profile”) lack an appropriate role such as “button” or “link.”

Language Lessons

The home screen of the app presents a list of languages, each leading to a list of modules. Expanding a module will reveal the

available lessons. VoiceOver does not indicate when a module has been expanded, so a user must explore the screen to find the list. This could be fixed by automatically moving the VoiceOver focus to the first incomplete lesson once a module has been expanded.

Unfortunately, much of the functionality of lessons is not accessible to VoiceOver users, as nothing happens when double-tapping on the name of a lesson.

The only way to access the audio files for a lesson is through the "Playlists" screen. From here, each lesson can be streamed or downloaded. The "Play" and "Download" controls do not have a button role.

Visual Appearance

Overall, Rocket Languages is a positive experience for low-vision testers. However, some improvements to the interface and providing appearance settings would go a long way toward making the experience better. The app's colour scheme is not adjustable, and colour and contrast frequently change from one screen to the next. An adjustable text size would further improve readability, but many icons are also small and difficult for testers to see, despite having enough space to be larger.

Development Recommendations

- Add roles such as "button" or "link" to actionable controls.
- Add labels to any controls that do not already have one.
- Where possible, make icons larger and easier to see.
- Improve the consistency of colour contrast across all screens.
- Allow for adjustment of visual appearance, particularly the font and colour scheme.
- Fix access to lessons from the modules list.
- Fix duplicate labels on text fields.

Rocket Languages App for Android

- Tested Android Version: 11 and 12
- Tested app version: 5.8.14

Signing In

The login screen has low contrast and does not support pinch-to-zoom. Low-vision testers found it unnecessarily difficult to navigate this screen.

Only one screen reader bug was found on the login screen. The "Remember Me" checkbox is not labelled. Screen readers identify it as a checkbox, but no further information is spoken.

General

The home screen shows a list of language courses, each expanding to reveal a list of lessons. Once a course is loaded, the resulting screen has several accessibility bugs.

The first button on this screen is not labelled. It opens a menu for navigating to the leaderboard and other sections of the app. Next, the "Options" control is not a button, though its label is clear.

When opening the "Options" menu, the menu items appear at the bottom of the screen but are not announced by a screen reader. Users need to find and move to these options manually. This should be a pop-up that takes focus away from the rest of the app until a menu item is selected.

On the "Profile" screen, each text field has a duplicated label before the field itself. This means that when a TalkBack user swipes right through the on-screen controls, the label will be read twice, followed by the text field in question. Each of these groups of three controls should ideally be merged into a single labelled text field.

In the "Settings" screen, users can set the difficulty level throughout the app. These selector buttons are missing a control type; consequently, TalkBack users are not told which of these difficulty levels is currently selected.

Language Lessons

Once a module is selected, a list of lessons for that module is shown. Selecting one of the lessons will show sections (such as 1.1, 1.2, etc.) TalkBack does not indicate that new content has loaded after expanding a module, lesson, or section.

Once a section has been selected, a "Start Lesson" button will appear. This control is not announced as a button by TalkBack. This issue also occurs if a lesson is partially complete and there is a "Continue" button instead.

The lesson screen has controls for playing the audio tracks and several learning activities. A number of notable accessibility issues were found on this screen.

The button to start audio playback is a blank clickable element with no label and control type. Screen readers will make a sound when moving to the control but will not announce anything. The label is a separate control, reachable by swiping left from the unlabeled button. The label itself cannot be double-tapped to start playback, so many users may not discover the blank control at all and will therefore be unable to start playback.

The "Download" control has a similar problem, the control itself is blank with no label, and the label is a separate control which cannot be double-tapped. To delete a download, it is necessary to press the "Download Options" button, which invokes a menu at the bottom of the screen. Like most other menus, this one appears without a screen reader announcement, and users need to explore the screen to find the options. The focus should instead be placed on the "Delete" option automatically.

Activities

Each language lesson has several interactive activities in addition to the audio track. All these activities suffer from a set of common accessibility bugs:

- The “Help” and “Close” buttons are blank controls with no label or control type. Screen readers read nothing when navigating to them.
- When pressing the “Help” button, a set of instructions is alternately shown and hidden. Screen readers do not indicate whether these instructions are expanded or collapsed, so the user must explore the screen to find them. In addition to being properly labelled, this button should indicate whether it is expanded, and the focus should be moved to the instructions when the button is pressed.
- The “Get Started” or “Continue” button does not have a control type.

In the “Play It” activity, both the “Help” and “Settings” buttons have no label and do not announce whether they are expanded or collapsed. When “Settings ” is expanded, the options appear at the screen's bottom without a screen reader announcement. The “Practice This Conversation,” “Stop,” and list of playable characters are all missing a control type, even though they are clickable. All these controls should be buttons.

Next to each character is a button that plays an example phrase spoken by that character. This control has no label and no control type. In the additional Vocabulary section, the “Play” and “Record” buttons are also blank controls with no label or control type. Whenever a phrase is played from any of these sections, TalkBack's focus is immediately moved to the top of the screen, making it necessary to find the correct spot on the screen manually.

Flashcards are unusable with a screen reader because the answer cannot be read once it is revealed. A workaround is to access settings and enable the “Play Audio on Reveal” setting. This still does not allow screen reader users to review the answer in text form, so this activity is not accessible. The “Settings,” “Flip,” and rating controls are missing a control type (button).

The “Hear It, Say It” activity is usable but with several unlabeled or blank controls. These include the “Record” and “Stop” buttons, the “Repeat” button, and the rating selection controls.

The “Know It” section asks the user to translate a phrase by speaking it. TalkBack does not read the phrase, so this section is completely unusable by screen reader users. TalkBack is also unable to read the correct answer after it is revealed. Like the previous section, the “Stop” and “Record” buttons are blank controls without a label or control type.

The quiz activity contains a “Play Quiz Audio” control. When swiping right to move through the controls, TalkBack reads the “Play Quiz Audio” label, but it cannot be double-tapped. The user must keep swiping and double-tap on the blank control instead. When audio is playing, the elapsed and remaining time controls are read correctly, but they need labels so users know which timestamp they are reading.

When a quiz has started, only one question is visually shown at a time, but TalkBack can navigate all the questions and associated answers at once. The quiz must always be completed in order by answering the first visible question and then pressing the “Reveal” button. A screen reader user who cannot see the screen may inadvertently navigate to other questions in the quiz and would then have no idea how this activity works. The information visible to TalkBack needs to remain consistent with the information shown on the screen.

The “Sort It” activity asks users to sort cards to match an auditory phrase. The play button for the auditory phrase is a blank control with no type and no label. More importantly, the cards are unreadable with TalkBack, and there is no feedback when moving them around the screen. This activity is completely unusable by screen reader users.

Visual Appearance

Overall, the app’s various screens have inconsistent contrast and colour, and no appearance settings are available. A simpler and

more consistent colour scheme with higher contrast would improve readability, and additional fonts would give users more control of text appearance.

Development Recommendations

- Label all controls throughout the interface so that screen reader users can identify them.
- Add control types to all controls. Screen readers and users will behave differently depending on whether a control is a button, a checkbox, a toggle switch, or just a piece of informative text.
- Improve contrast and readability throughout the interface. Add text appearance settings.
- Ensure the source and translation text within activities can be read by TalkBack.
- When a menu is expanded, move the TalkBack focus to the menu options. Buttons which toggle a menu should also have a label which indicates whether the menu is currently shown.

Conclusion

The Rocket Languages apps and website contain many accessibility barriers, making the service difficult or impossible to use when working with assistive technologies. For low-vision testers, the interface was inconsistent and, at times, difficult to read on all tested platforms. For users relying on screen readers, some of the core functionality was utterly unusable, particularly in the mobile applications, and many other bugs compounded to create a frustratingly inconsistent experience.

Systems and Assistive Technology

- Operating Systems
 - Windows 10 (21H1)
 - iOS 14+
 - Android 11 and 12
- Mobile Applications

- Rocket Languages 5.6.5 (iOS)
 - Rocket Languages 5.8.14 (Android)
- Browsers
 - Chrome 86, 91
- Screen-readers
 - NVDA 2021.1 (Windows)
 - JAWS 2021 (Windows)
 - VoiceOver Touch (iOS)
 - TalkBack (Android)
- Magnification
 - Magnifier (Windows)
 - ZoomText 2020 (Windows)
 - Zoom (iOS)

Acknowledgements

The following testers and editors contributed to this report:

- Simon Jaeger (Lead Writer)
- David Kopman
- Riane LaPaire
- Ka Li
- Laetitia Mfamobani
- Deanna Ng
- Megan Sellmer

Published by the [National Network for Equitable Library Service](#) (NNELS), Vancouver, BC, 2022