

# 視覚障害者に対応した Webブラウザ上の 電子書籍リーダー試作

2025年2月25日 株式会社ボイジャー

北原 昌和

## Webブラウザとスクリーンリーダーを組み合わせることによる問題点

- iOS・iPadOSのVoiceOverやAndroidのTalkBackを有効にすると、タップやスワイプといった操作が通常の操作とは大きく変わる
  - 要素をタップすると、その要素を選択する
  - 画面のどこでも良いので、左右フリックすると、選択中の要素が切り替わる
  - 画面のどこでも良いので、ダブルタップすると、選択中の要素が決定する
- タップやスワイプ・ドラッグといった操作を前提としたプログラムは動作しない場合がある

## Webブラウザとスクリーンリーダーを組み合わせることによる問題点

- スクリーンリーダーがショートカットキーの制御を優先して使ってしまう
  - スクリーンリーダーが使っているショートカットキーは、JavaScriptのプログラムには通知されなくなり、ショートカットキーに割り当てていた機能が使えなくなる
- スクリーンリーダーとJavaScriptのインターフェースが無い
  - タッチ位置を取得できない
  - 任意のテキストをスクリーンリーダーで読み上げることができない
  - スクリーンリーダーが読み上げている状態かを把握できない
  - スクリーンリーダーが動作しているかを把握できない



# スクリーンリーダーで操作可能なユーザーインターフェースの実装

再生 または 停止	前のページ	次のページ	前章	次章	目次	現在位置
本文検索	しおり	クイック設定	設定	サイドメニュー	本を閉じる	
パネルを閉じる						

弊社のWeb上で動作する電子書籍リーダーであるBinBにスクリーンリーダーで操作可能な操作パネルを実装した。

代真つ只中の方たちに向けて、デジタル出版での取り組み方まで書いた作品です。50代は社会が興味を持つべきです。本を書くために必

多持っています。その経験談を

継ぐことは、次の10年、20年を充実させる

か？ 本なのか？ すでにブログや「F a

（エクスブック）」、「X（エックス、旧T

にアカウントを持つておられるなら疑問に

NSは自分が思い付いたことを手軽に発信

利な方法です。一方、経験をまとめるの

瞬間的につぶやくのはいいのですが、海

の数秒で他の発信に埋もれていって、集

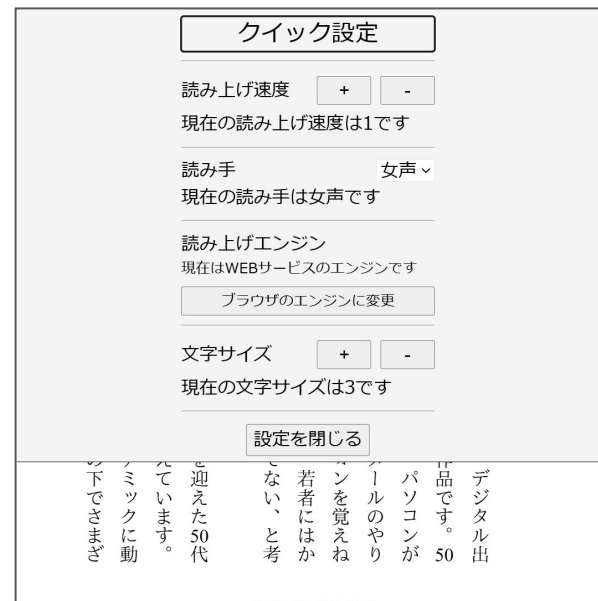
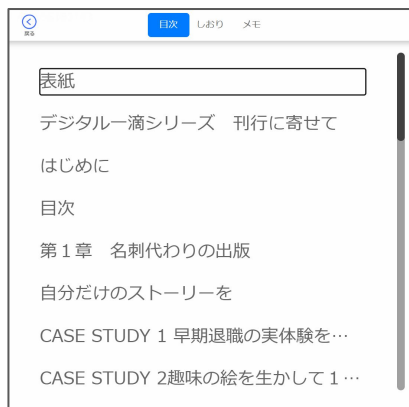
です。自分の経験を形として残すには本が

ます。デジタル技術が全てを変えました。

電子書籍の基礎を理解すれば、誰でも本を

- PC-Talker Neo、iOS・iPadOSのVoiceOver、AndroidのTalkBackに対応
- スクリーンリーダーが有効な環境でも確実に操作できるようにフォーカス可能なボタンで構成
- スクリーンリーダーを使わずにコンテンツを読み上げる機能を実装(コンテンツを保護する必要があるため)

# スクリーンリーダーで操作可能なユーザーインターフェースの実装



目次画面や本文検索画面などの既存の画面も、各項目をフォーカス移動で選択できるように変更

検索画面は使い勝手が悪いため、読み上げに関する項目のみに絞ったクイック設定画面を追加



## 開発後の視覚障害者のご意見

- リーダー起動時に音声でアナウンスがあるので安心感がある
- リーダーを操作して読み上げを行うことはできる
- 設定画面が使いにくい
  - 項目数が多いために必要な項目に行き着くのが大変
  - 現在設定されている値がわからない
- タッチイベント等が利用できないために実現できないこと
  - VoiceOverやTalkBackのジェスチャー操作に対応してもらいたい
    - 「戻る」操作や「読み上げ開始」「読み上げ停止」等
  - 詳細読みを使いたい
    - 詳細読みは文字を1文字ずつ確認するとき使用する機能。
  - コンテンツの内容をコピーしたい
  - コンテンツ内のリンクが利用できない

## サービスを提供するにあたってリーダー以外に対応しなければいけないこと

- 実際にユーザーがコンテンツを閲覧するには、
  - 読みたいコンテンツを探す
  - ユーザー登録する
  - 購入する
  - 購入したコンテンツを開くといった事が必要になる。
- ストア等のサービスのWebページもアクセシビリティ対策が必要
  - **WCAG**や**JIS X 8341-3:2016**等を考慮した設計をするのが理想

## Webページ アクセシビリティ対策例

- タグとCSSでデザインする
  - 文字間隔を開けるために空白文字を入れたり、文章の途中で改行するといったレイアウト調整はしない
- 適切に見出し(hタグ)を使いページを構成する
  - 全体像の把握や、選択する際のスキップ等に効果がある
- 画像にはALT属性で説明を入れる
- ログインボタン・閲覧ボタンのような機能的な要素は必ずフォーカスできるようにする
- キーボードだけですべて操作できるようにする

## Webページ アクセシビリティ対策例

- 要素の順番を意識する
  - 検索語句入力欄と検索実施ボタンを並べる
  - ログイン等使用する機会が高いものはページの先頭の方に配置する
  - ボタン等の説明をその要素の後ろに配置しない
- リンクは、リンク先が何かを把握できるように書く
- ブラウザの拡大を行っても対応できるように意識する(200%表示には耐えられる程度は必要)
- 文字やボタンの色は背景と十分なコントラスト比があるようにする
- 色や形、位置などの表現で指定しない
  - 「赤の項目は必須です」「右の項目は」などの表現を使わない