Engineer

スクリーンリーダー

ページの内容を音声で読み上げたり、点字で提示したりするソフトウェア

どのようなものか

スクリーンリーダーは、コンピューターやスマートフォン上のあらゆる内容を、合成音声や、表示を更新できる点字ディスプレイ上の1行の点字として提示するプログラムです。文書の本文から、その周囲にあるボタンと各ボタンの状態までを伝えます。利用者は、キーボードやタッチジェスチャーでそのインターフェースを操作できます。一つの製品名ではなく、さまざまなスクリーンリーダーを指す総称です。Windows向けには、無料でオープンソースのNVDAのほか、JAWSもあります。VoiceOverはmacOSとAppleのモバイルOSに組み込まれており、AndroidにはTalkBackがあります。利用者の多くは全盲やロービジョンの人ですが、画面を見ることができても、読むより聞く方が楽なために使う人もいます。

開発者が見落としやすいのは、読み上げる言葉がどこから来るかです。スクリーンリーダーは画素を見ているわけではありません。ブラウザーはHTMLを、アクセシビリティツリーという別の構造へ変換します。この構造では、各要素を役割（ボタン、リンク、見出し、リスト）、名前（その要素を識別する言葉）、状態（チェック済み、展開済み、無効）に整理します。OSのアクセシビリティインターフェースがこのツリーをスクリーンリーダーに渡し、それが読み上げられます。見た目は紙飛行機の絵でも、コードにテキストのないボタンは、このツリーでは名前のないボタンになり、読み上げもそのとおりになります。

また、利用者はページを上から下まで順に聞くわけでもありません。NVDAのブラウズモードでは、1文字のキーを移動に使えます。`H` は次の見出し、`D` は次のランドマーク、`K` は次のリンク、`F` は次のフォーム項目へ移動し、1から9の数字はそのレベルの見出しへ移動します。`NVDA+F7` で、ページ内のすべてのリンク、見出し、ランドマークの一覧を開けます。VoiceOverには `VO-U` で開くローターがあり、同様の一覧を使えます。目で読む人が太字や余白に目を走らせて拾い読みするように、スクリーンリーダーの利用者は構造上の要素を飛び移って拾い読みします。その構造のないページには、移動先がありません。

概念を判断に生かす

「スクリーンリーダーとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 読み込みが速く、スマートフォンでも使え、お客様がどのブラウザーを使っていても同じように動く画面を提供します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 フルスタックのGIS地図開発を指揮 — DBからブラウザーまでのフルスタック地図開発

フルスタックのGIS地図開発を指揮 お問い合わせ →

なぜ重要なのか

チームが実際に判断すべきなのは、公開するインターフェースが、マークアップの表す情報だけを受け取る人にも使えるものになっているかどうかです。スクリーンリーダーほど、ページを記述どおりに受け取るものはありません。レイアウト、色、位置、誰もが知るアイコン、赤くなる入力欄など、見た目にしか存在しない手掛かりはすべて失われ、コードが表すことだけが聞き手に届きます。

そのため、どれほど画面を眺めても、このマークアップの検査に代えることはできません。デザインレビュー、スクリーンショットの比較、慎重なクリック操作の確認は、いずれも視覚を使った確認です。スクリーンリーダーが明らかにする欠陥は、そもそも視覚では分からないものです。自動のルール検査も役立ち、代替テキストのない画像、ラベルのないフォーム部品、取り得ない値を持つ属性など、実際の問題をまとめて見つけます。ただし、検査するのは名前が存在するかであって、それが適切な名前かではありません。購入手続きを進めるボタンが「ボタン」、メニューが「ハンバーガー」と読み上げられることに気付くには、人が数分間聞く必要があります。

その人手と数分の時間にかかる負担は、多くのチームが考えるほど大きくありません。NVDAのインストールは無料で、社内のMacやiPhoneにはすでにVoiceOverが入っているため、最初のテストに購入や専門家は必要ありません。誰かが目を閉じるか画面を消し、キーボードと耳だけでページの主な作業を完了できるか試せばよいのです。早い段階で行えば、見つかった問題の修正はたいていさらに安く済みます。標準の `button` 要素や `a` 要素には役割とキーボード操作が備わっていますが、クリック可能にした `div` では、それらをすべて手作業で実装し、保守しなければなりません。

一方で、スクリーンリーダーごとに違いがあります。アクセシビリティツリーの解釈は製品ごとに異なり、ツリーの構築もブラウザーごとに異なるため、ある組み合わせでテストしたページが、別の組み合わせでは違う読み上げになることが

あります。一つの組み合わせを徹底的にテストするより、よく使われる二つの組み合わせでテストする方が、確認できる範囲は広がります。

実際の活用例

あるアプリケーションでは、欠陥は聞けばすぐに分かるものでした。操作部品はシステムのシンボルで描かれていましたが、アクセシビリティラベルのないシンボルは内部名で読み上げられるため、送信ボタンは「arrow up circle, button」と名乗っていました。アイコンだけの17個の操作部品に、その機能を表す名前を付けました。形ではなく操作内容を表すラベルにし、状態によって意味が変わる部品では、ラベルも状態に追従させました。現在は、アイコンだけの操作部品から8行以内にアクセシビリティラベルがあることをリンターが要求するため、ラベルのない18個目は追加できません。この距離は、実測に基づいて選んだ、意図的に粗い判定基準です。事例ページはその限界も明記しています。このルールが証明するのは近くにラベルがあることだけであり、そのラベルが正しいかは、今も人が聞いて確認する必要があります。

文書でも、形式を変えて同じ問いが生じます。画面上で正しく見えるPDFでも、スクリーンリーダーで読めるとは限りません。そこで、記録に残る検証段階では、各言語にまたがる履歴書の各版、リファレンスシート、ポートフォリオ、カバーレターという9つの生成文書を、PDF/UAアクセシビリティプロファイルの独立した検証ツールで検査し、設定されたアクセシビリティプロファイルの106ルールすべてに合格しました。合格には、タグ付きの構造、文書言語の宣言、装飾でないすべての画像の代替テキスト、メタデータ内の適切なタイトル、そしてレイアウトから偶然生じる順序ではなく明示的な読み上げ順序が必要でした。二つの数値はいずれも一つの検証ツールによるものであり、事例ページにもそう記されています。

このサイトでも、自動監査を同じように扱っています。クレジットページでは、監査を証明ではなく最低限の基準と位置付けています。また、このサイトをスクリーンリーダーで利用していて読み上げの誤りに気付いた人には、知らせてほしいと呼び掛けています。それは、品質ゲートでは生成できない報告だからです。

問題が起きるところ

スクリーンリーダーの問題の多くは画面では分からず、そのためレビューをすり抜けます。よくあるのは、次のような形です。

操作部品が自分の絵柄を読み上げます。 テキストのないアイコンボタンは、コードが与えるファイル名やシンボル名などで名付けられます。何もなければ、単に「ボタン」と読み上げられます。アイコンだけの操作部品にはすべて、操作内容を表す名前を付け、絵そのものはツリーから隠します。

```
<button type="button" aria-label="Send message">
    <svg aria-hidden="true" focusable="false"><!-- icon --></svg>
</button>
```

ページが一続きの文章として読み上げられます。 見出しを装った `div` 要素や、ランドマークのないレイアウトは、見た目には問題なくても、 `H` や `D` で移動できる先を残しません。利用者は目的の箇所を探すために、矢印キーで1行ずつ進まなければなりません。本来の `h1` から `h6` の要素を順序どおりに使い、ページの各領域には `header` 、 `nav` 、 `main` 、 `footer` を使います。

何かが変わっても、聞き手には何も伝わりません。 入力欄の下にエラーメッセージが現れ、買い物かごの件数が増え、トースト通知が現れては消えます。目で見ている利用者には分かっても、フォーカスが別の場所にあるため、スクリーンリーダーは何も言いません。メッセージへフォーカスを移すか、動的な更新を `aria-live` で指定した領域に置き、読み上げられるようにします。

隠した内容が読まれたり、見える内容が読まれなかったりします。 CSSで画面外へ押し出したテキストは、アクセシビリティツリーには残り、読み上げられます。見える要素に `aria-hidden="true"` を指定すると、他の人には見えたままでも、聞き手には届かなくなります。どちらも意図して使えば役立ちますが、誤って使うと混乱の原因になるため、画面だけでなくツリーに何が含まれるかを確認します。

すべての検査に合格しても、読み上げは不適切です。 ラベルがあっても間違っていることはあり、代替テキストが画像の要点ではなくファイルを説明していることも、読み上げ順序が形式上は正しくても意味をなさないこともあります。構造を変更したページには、公開前のチェックリストに短時間の読み上げ確認を加えます。

参考資料

- ツールと技法 — 障害のある人がウェブをどう利用するかを説明するW3Cの概説です。他の支援ツールとともに、スクリーンリーダーも扱っています。

- NVDAユーザーガイド — 無料のスクリーンリーダーの開発元による、ブラウズモード、1文字キーによる移動、要素リストの説明です。
- MacでVoiceOverローターを使う — 見出し、リンク、ランドマークごとの移動方法を説明するAppleのガイドです。
- Core Accessibility API Mappings 1.2 — 役割、名前、状態が、ブラウザーからスクリーンリーダーの読み取るプラットフォームのインターフェースへどう渡されるかを説明しています。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「スクリーンリーダーとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く 連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/screen-reader/