

山口県モバイルアプリケーションアクセシビリティガイドライン

令和6年（2024年）3月

山口県総合企画部デジタル推進局
デジタル・ガバメント推進課

目次

1. はじめに	1
(1)ガイドラインの目的.....	1
(2)モバイルアプリのアクセシビリティの確保.....	1
(3)ガイドラインの使い方.....	2
2. 具体的な配慮項目.....	3
(1)基本的な画面設計	3
[01]表示の向き	3
[02]リフロー(画面幅に合わせた表示).....	4
[03]画面配色変更をしている利用者への対応.....	5
(2)構造及び表示スタイル.....	6
[04]文書の構造化を図る.....	6
[05]表示順に意味がある場合の配慮.....	10
[06]タイトルはアプリ名や機能がわかるものにする.....	11
(3)操作および入力.....	12
[07]デザインの統一と操作性.....	12
[08]コンテンツのスキップ、メニューの読み飛ばし.....	13
[09]全ての操作がキーボードだけで可能であること.....	14
[10]キーボードショートカット設定時のルール.....	16
[11]簡単なジェスチャでの操作.....	17
[12]操作のキャンセル.....	18
[13]動きによる起動.....	19
[14]入力欄につける説明.....	20
[15]入力欄のエラー回避・対応.....	21
[16]入力補助.....	22
[17]画像などを用いた認証使用時の配慮.....	23
[18]アクセシブルな認証.....	24
[19]入力の時間制限.....	25

[20]変化・追加されるテキストメッセージを通知する	26
[21]自動で追加表示されるコンテンツへの配慮(キーボード・マウス操作時).....	28
[22]自動的な画面遷移を避ける.....	29
[23]操作ボタン、リンク設定のルール.....	30
[24]操作ボタンなどのサイズ.....	31
(4)非テキスト情報.....	32
[25]画像の代替テキスト.....	32
[26]音声の代替テキスト.....	33
[27]映像の代替テキスト.....	33
[28]字幕・キャプション.....	34
[29]動画の音声解説.....	35
(5)色及び形.....	36
[30]色や形など感覚的な特徴だけで情報を区別しない.....	36
[31]文字色と背景色の組み合わせ	37
[32]文字以外の意味のあるコンテンツと周囲の色の組み合わせ	38
(6)文字.....	39
[33]画像化された文字の使用制限.....	39
[34]位置調整目的のスペースの禁止.....	40
[35]テキストの間隔.....	41
(7)音.....	42
[36]音を使用する際の注意.....	42
(8)速度.....	43
[37]変化・移動する画像や文字の禁止.....	43
[38]激しい画面点滅の禁止.....	43
(9)言語.....	44
[39]言語の指定.....	44

1. はじめに

(1) ガイドラインの目的

本ガイドラインは、山口県モバイルアプリケーションアクセシビリティ方針の基、本県が制作し、提供するモバイルアプリケーション（以下、「モバイルアプリ」という。）について、年齢や障害の有無などに関わらず様々な人が利用しやすいよう、モバイルアプリを制作する際などに留意が必要な事項をまとめています。モバイルアプリ全体のアクセシビリティを向上させるために、モバイルアプリの導入や運用に際して、このガイドラインを必ず確認するようにしてください。

(2) モバイルアプリのアクセシビリティの確保

モバイルアプリは、主にモバイル端末の画面をタッチして操作します。そのため、視覚障害者が画面情報を取得できなかつたり、手の不自由な人が操作しづらいなど、利用者が障壁（以下、「バリア」という。）を感じる場合があります。様々な人が利用することを想定し、そのニーズに合わせてアプリを提供することが必要です。一方で、アクセシビリティの対象は、現在も主にウェブサイトが中心であり、モバイルアプリについての一般的なガイドラインがありません。

本ガイドラインでは、国際的なウェブサイトのアクセシビリティのガイドラインである WCAG2.2 を参考に留意すべき項目をまとめています。

なお、モバイルアプリの種類（※1）や技術的制約（※2）などによっては、取り入れることが難しい項目があります。制作するモバイルアプリに合わせて必要な配慮内容を検討してください。

※1 モバイルアプリの種類

ネイティブアプリ：OS 別にプログラミングされたアプリ。インストールして使用。

ウェブアプリ：OS によらずブラウザ上で動くアプリ。HTML や JavaScript で作成。

ハイブリッドアプリ：ネイティブアプリの一部の機能として OS のブラウザを利用するアプリ。
ウェブアプリの機能とネイティブアプリの機能を持つ。

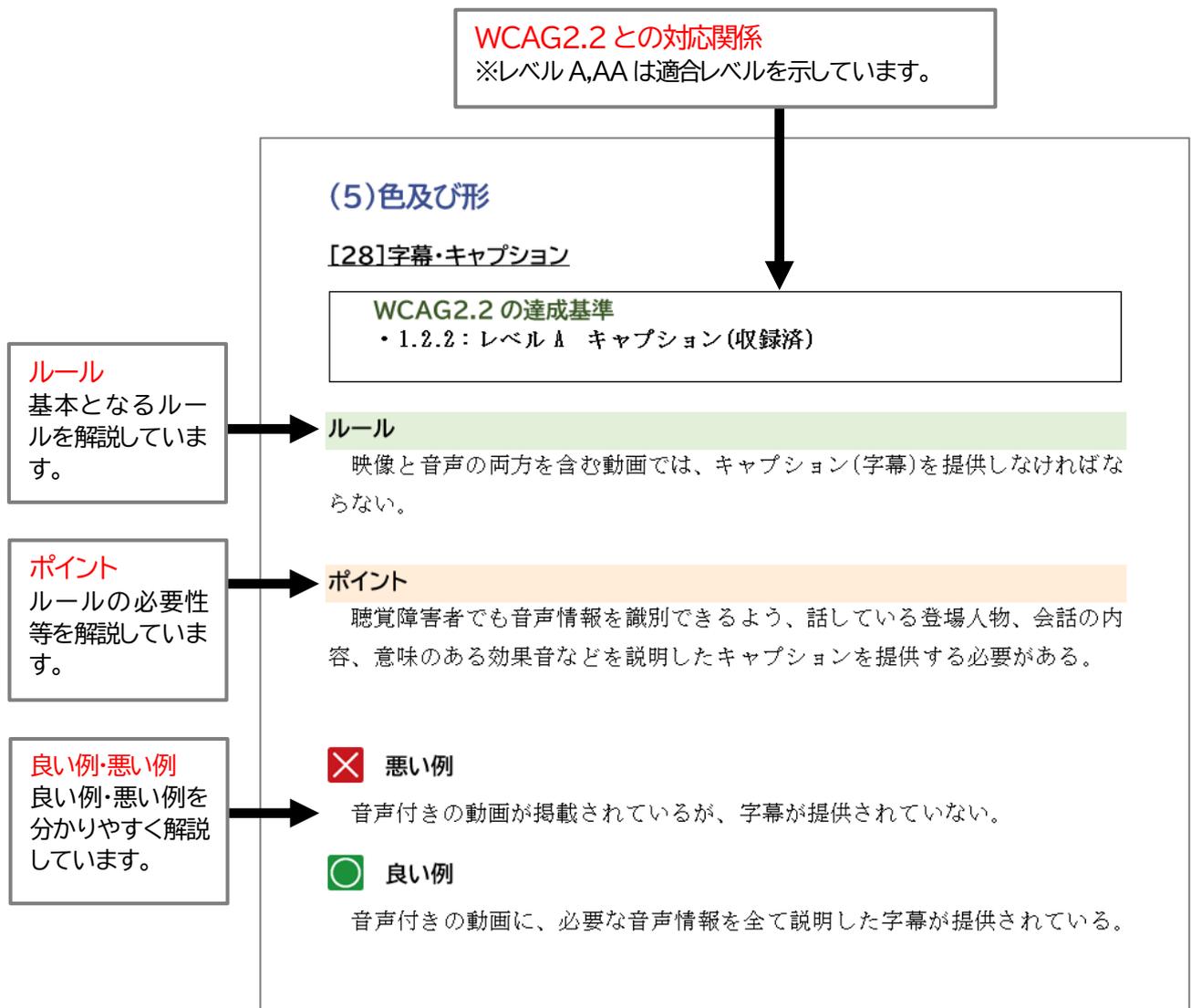
※2 技術的制約

モバイルアプリ制作の技術的な内容については各 OS の情報を参照ください。

iOS：<https://developer.apple.com/jp/accessibility/resources/>

Android OS：<https://developer.android.com/guide/topics/ui/accessibility?hl=ja>

(3)ガイドラインの使い方



2.具体的な配慮項目

(1)基本的な画面設計

[01]表示の向き

WCAG2.2 の達成基準

- 1.3.4 : レベル AA 表示の向き

ルール

表示の固定が必要不可欠な場合を除き、画面の向きを制限することなく、縦長・横長のどちらの向きにしても、表示・操作ができるようにする。

ポイント

このルールは、常に片腕で端末を利用している場合やスマートフォンホルダー等の器具で端末を車椅子に固定している場合等、容易に端末の向きを変えることのできないユーザーが、端末の向きを変更せずコンテンツを利用できるようにするためのものである。

[02]リフロー(画面幅に合わせた表示)

WCAG2.2 の達成基準

- 1.4.4 : レベル AA テキストのサイズ変更
- 1.4.10 : レベル AA リフロー

ルール

文字サイズを変更できる場合は、200%（縦・横それぞれ2倍）まで拡大した場合でも、他のコンテンツを隠すことなく、閲覧や操作ができるよう作成する。

また、その際、コンテンツは、画面幅に合わせて折り返し表示されるように作成し、必要な場合を除き、水平方向と垂直方向のスクロールバーを同時に表示させることなく、支障なく利用できなければならない。

ポイント

このルールは、主に弱視のユーザーがコンテンツを拡大表示して利用する場合に、他のコンテンツが隠されたり、コンテンツ同士が重なって読めなくなることを避けるためにある。

また、画面幅により折り返し表示することにより、余計なスクロールなしにコンテンツを読むことができるようにするためのものである。

水平方向・垂直方向のスクロールバーが両方必要となるケースとして、大きなサイズの画像・表、動画、地図を表示する、といったものは、コンテンツの理解のためにスクロールが必要なため対象外となる。

[03]画面配色変更をしている利用者への対応

WCAG2.2 の達成基準

・1.1.1：レベルA 非テキストコンテンツ

ルール

OS の機能で配色を変更している利用者が問題なく操作できるようモバイルアプリを設計する。

ポイント

通常の画面配色では見づらいつと感じる利用者が、OS の機能を使って配色を変更することがある。弱視の利用者の中には、明るい背景では眩しく見づらいつことから黒色背景に白色の文字に変更（ハイコントラストモード、またはダークモード）する人がいる。そのため、モバイルアプリが対応していない場合、コンテンツの内容を把握できずアプリを利用できなくなる。

- ※ OSによっては、利用者の配色設定によって、アプリの配色を変更する機能があるため、その機能を使用することが有効である。
- ※ アプリ内のウェブ部分では、意味のある画像をCSSで背景画像として設定していると表示されないことがあるため注意が必要である。

悪い例

ハイコントラストモードに設定したところ、見えなくなってしまう文章やボタン、画像がある。

良い例

ハイコントラストモードに設定すると、背景が黒、文字が白となり、見えなくなる文章やボタン、画像はない。

(2)構造及び表示スタイル

[04]文書の構造化を図る

WCAG2.2 の達成基準

- 1.3.1 : レベル A 情報及び関係性
- 2.4.6 : レベル AA 見出し及びラベル

見出し・箇条書き・表・表題等文章の構造を適切に表現する必要がある。その場合、デザインやレイアウトのみを調整するのではなく、適切な技術を使い設定する。また、テキストボックスやチェックボックス、ラジオボタン等を設置する場合は、その説明のテキストと関連付けを行う。

以下については、ハイブリットアプリのウェブ部分の例であるが、ネイティブアプリにおいても文章構造を明示できるのであれば使用する。

《見出し》

ルール

外見上、または文脈上見出しとして捉えられるテキストは、見出しであることを明示する。

HTML では、見出し要素(「<h1>」～「<h6>」)を設定する。

ポイント

スクリーンリーダーの利用者は、見出しを手掛かりに必要な情報を探ることが多い。そのため見出しと明示されていない場合、情報を探ることが困難になる。

参考：iOS の VoiceOver 利用の場合、「ローター」ジェスチャーを使うことで見出し等、利用者が移動したい単位で移動することができるようになる。

iPhone、iPad、iPod touch の VoiceOver ローターについて

<https://support.apple.com/ja-jp/111796#:~:text=VoiceOver>

❌ 悪い例

- 文字サイズを大きくすることだけで見出しとしている。太字や背景色の違いだけで見出しとしている。

○ 良い例

- 文字サイズを大きくして見出しであることを示している部分に、プログラムとしても見出しであることが明示されている。(HTML ではh 要素)
- 太字や背景色の違いによって見出しであることを示している部分に、プログラムとしても見出しであることが明示されている。(HTML ではh 要素)

《箇条書き》

ルール

箇条書きの部分は、箇条書きであることを明示する。

HTML では、3 種類の箇条書きの要素が用意されているため、箇条書きの種類に応じて適切なリスト要素を設定する。

「・」等が振られた順序なしリストであれば「」要素と「」要素、番号が振られた順序付きリストであれば「」要素と「」要素、言葉の説明等を列記する定義リストでは「<dl>」・「<dt>」・「<dd>」要素等。

ポイント

スクリーンリーダーでは、箇条書きであること。また、その項目数を音声で知らせるものがある。箇条書きであることを適切に示すことは、スクリーンリーダーの利用者がコンテンツの内容を把握するために有用である。

✕ 悪い例

文頭に「・」を付けているだけで、箇条書きにリスト要素が設定されていない。

○ 良い例

箇条書きの種類に応じて適切なリスト要素が設定されている。

例：順序なしリスト

```
「<ul>  
<li>リンゴ</li>  
<li>バナナ</li>  
<li>みかん</li>  
</ul>」
```

《表》

ルール

縦横の並び順に意味のある表では、最初の行や、左の列を表見出しとして設定する。
HTML では「<th>」要素を設定する。

ポイント

スクリーンリーダーを利用して表を閲覧する際、「<th>」要素が設定されたセルの内容は行・列を移動する度に読み上げられるようになっている。

そのため、見出しとして捉えられるセルに「<th>」が設定されていない場合や、見出し以外のセルに設定されている場合には、情報を理解しづらくなる。

悪い例

見出し列に「<th>」要素が設定されていない。

```
「<table>  
<tr><td>4 月</td><td>5 月</td><td>6 月</td></tr>  
<tr><td>1200</td><td>1480</td><td>1120</td></tr>  
</table>」
```

良い例

見出し列に「<th>」要素が設定されている。

```
「<table>  
<tr><th>4 月</th><th>5 月</th><th>6 月</th></tr>  
<tr><td>1200</td><td>1480</td><td>1120</td></tr>  
</table>」
```

《表題》

ルール

表の直前に表題がある場合は、表と関連付け、表題であることを明示する。
HTML では、見出し要素または「<caption>」要素を設定する。

ポイント

表題に見出しやキャプションを表す要素が設定されていない場合、特に表題と表本体が離れている場合等、スクリーンリーダーではテキストが表題であるかを識別しづらくなる。

✖ 悪い例

表題が「<p>」要素を使って記述されている。
「<p>上期売り上げ表</p>
<table>
・・・」

○ 良い例

表題に見出し(「<h2>」)要素が設定されている。
「<h2>上期売り上げ表</h2>
<table>
・・・」

[05]表示順に意味がある場合の配慮

WCAG2.2 の達成基準

- 1.3.2 : レベル A 意味のあるシーケンス
- 2.4.3 : レベル A フォーカス順序

ルール

箇条書きや表を活用して、文章をきちんと構造化し、スクリーンリーダーの読み上げや、キーボード操作時でも、コンテンツが意味のある順番に並ぶようにする。

ポイント

見た目上の画面配置が整っていても、スクリーンリーダーの読み上げで制作者が意図しない順番で読み上げたり、キーボード操作時にキーボードフォーカスが想定していない順番で動くことがある。見た目の配置だけでなく、読み上げの順番、キーボードフォーカスの順番に配慮が必要である。

悪い例

- 左に東日本地域の県名一覧、右側に西日本地域の県名一覧があり各県のウェブサイトへリンクされている。スクリーンリーダーの読み上げは、左右のリストを交互に読み上げ、キーボードフォーカスもその順序で移動する。
- モバイルアプリの設定方法が縦2列・横2列で4つの手順に分けて書かれている。スクリーンリーダーでは、手順1を読み上げた後、右に書かれている手順3を読み上げる。手順2は、手順1の下に書かれている。

良い例

- 左に東日本地域の県名一覧、右側に西日本地域の県名一覧があり各県のウェブサイトへリンクされている。それぞれの一覧は箇条書きとしてまとめられており、スクリーンリーダーの読み上げは、東日本の一覧をすべて読み上げた後に西日本の一覧を読み上げ、キーボードフォーカスもその順番に移動する。
- モバイルアプリの設定方法が縦2列・横2列で4つの手順に分けて書かれている。並び順は悪い例で示した並びであるが、スクリーンリーダーでは、手順通りの順番で読み上げられる。

[06]タイトルはアプリ名や機能がわかるものにする

WCAG2.2 の達成基準

- 2.4.2 : レベル A ページタイトル

ルール

画面にタイトルをつけることができる場合は、モバイルアプリの名前や画面の機能を示すタイトルをつける。

ポイント

画面が切り替わった時にタイトルでその画面の機能が識別できれば、利用者が画面全体を確認することなく、安心して操作できる。特にスクリーンリーダーの利用者にとって有用である。

(3)操作および入力

[07]デザインの統一と操作性

WCAG2.2 の達成基準

- 3.2.3 : レベル AA 一貫したナビゲーション
- 3.2.4 : レベル AA 一貫した識別性
- 3.2.6 : レベル A 一貫したヘルプ

ルール

複数の画面で表示される同じ意味を持つボタンや画像は、統一したデザインとする。また、複数画面で表示される他画面への移動ボタンがある場合は、デザインだけでなく位置や順番もモバイルアプリ内で統一する。

ポイント

モバイルアプリ内で統一されたデザイン・配置は、利用者がスムーズにアプリを操作することに役立つ。

特に、FAQ 等利用者が操作に迷った際、ヘルプ機能として役立つものについては、どの画面でも同じ位置に配置し、簡単に見つけられるようにする。

[08]コンテンツのスキップ、メニューの読み飛ばし

WCAG2.2 の達成基準

・2.4.1：レベルA ブロックスキップ

ルール

画面上部に共通して表示される操作ボタン、共通メニュー等があり、キーボード操作で画面固有の機能部分に移動するために多くのキー操作が必要な場合は、スキップできるようにする。スキップの方法としては、見出しをつけることが有効である。

(参照：本ガイドライン[\[04\]文書の構造化を図る](#))

ポイント

キーボード操作では、画面の最上部から順番にキーボードフォーカスを移動させ操作していくため、上部にある画面共通の操作ボタンやリンクがあると、重要な画面固有の情報や操作部分にたどり着けない、時間を要することがある。

- ※ ウェブサイトでは「ページの先頭に本文へジャンプするページ内リンクを設置する方法」が多く実装されている。画面共通の部分が多い場合は、同様の仕組みも有効である。

悪い例

画面共通のボタンをスキップする機能がないため、キーボード操作で、画面固有の情報・機能にたどり着くまでに時間を要する。

良い例

各画面の固有の情報の最初に見出しが設定されているため、見出しジャンプ機能で素早く情報にたどり着くことができる。

[09]全ての操作がキーボードだけで可能であること

WCAG2.2 の達成基準

- 2.1.1 : レベル A キーボード
- 2.1.2 : レベル A キーボードトラップなし
- 2.4.7 : レベル AA フォーカスの可視化
- 2.4.11 : レベル AA 隠されないフォーカス (最低限)

ルール

タッチ操作で操作できるコンテンツは、全てキーボードでも同様にフォーカスが当たり、選択・操作できなければならない。そのためには、キーボードフォーカスがどの操作部分に当たっているか、他のコンテンツに隠されることなく判別できるようにする。

また、同様の観点から、キーボード操作(Tab キー等)でフォーカスを移動する際、特定の箇所から先に進まない、一定の範囲でループするといった状況にならないようにする。

ポイント

タッチ操作が困難な利用者は、キーボードを使用してモバイルアプリを利用することがある。そのような利用者が支障なく利用できるようにするための配慮である。

フォーカスが当たっている部分を判別させる方法として、点線で囲む、色を変化させる等がある。

悪い例

- キーボード操作の際、フォーカスが当たらないボタンがある。
- Tab キー操作の際、特定の箇所から先に進めなくなる。
- キーボード操作の際、どこのボタンにフォーカスが当たっているのか判別できない。
- キーボード操作でフォーカスが当たった際、表示された画像により、キーボードのフォーカスが当たっている部分が隠れてしまい、その後の操作が困難となる。

良い例

- リンクやボタン等操作可能な部分は、すべてキーボード操作でフォーカスが当たり、操作可能となっている。
- Tab キーで操作している際、特定の箇所では動かないことがなく、すべての操作が可能となっている。
- キーボード操作の際、フォーカスが当たっている部分が枠線で囲まれて表示される等、フォーカスが当たるすべての部分で、どの部分にフォーカスが当たっているのか判別できる。
- キーボード操作でフォーカスが当たった際に画像が表示されるが、キーボードのフォーカスを隠すことがない。

[10]キーボードショートカット設定時のルール

WCAG2.2 の達成基準

・2.1.4：レベルA 文字キーのショートカット

ルール

モバイルアプリ・ウェブサイト内で単一キー(アルファベット、数字、記号、句読点等)のキーボードショートカットを設定することは避ける。やむを得ず設定する場合、利用者が下記のいずれかを利用できるようにする。

- ショートカットキーを無効化することができる。
- 特殊キー(Windows キーボードの場合：Ctrl、Alt、Win 等)との組み合わせに変更できる。

ポイント

スクリーンリーダーには各キーに独自の機能が設定されており、モバイルアプリ側でショートカットキーを割り当てた場合、スクリーンリーダーの機能を活用できなくなる。

また、キーボード利用者にとっても誤ってショートカットを呼び出してしまう危険性があるため、1文字のキーに操作を割り当てないようにすることが重要である。

悪い例

メインメニューを展開するためのショートカットキー「m」が設定されているにも関わらず、それを無効化・変更する手段が用意されていない。

良い例

メインメニューを展開するためのショートカットキー「m」が設定されているが、ページ最上部にその機能を無効化するためのボタンが設置されている。

[11]簡単なジェスチャでの操作

WCAG2.2 の達成基準

- 2.5.1 : レベル A ポインタのジェスチャ
- 2.5.7 : レベル AA ドラッグ動作

ルール

1 本指での簡単なジェスチャで操作できるようモバイルアプリを設計しなければならない。

複雑なジェスチャに機能を持たせる場合でも、簡単なジェスチャで同じ機能が使えるようにする。

- ※ 2本以上の指を使用するタッチ操作、指を左右に素早く動かす等、指の軌跡に依存する操作は複雑なジェスチャに含まれる。マウス操作でドラッグする操作も複雑な操作である。

ポイント

タッチ操作のみに依存してモバイルアプリを設計した場合、特別な入力機器を使用している利用者が操作できない可能性がある。例えばタッチペンを利用している利用者は2本指でのジェスチャを再現できない。そのような利用者もモバイルアプリの機能を活用できるよう設計する。

また、マウスのドラッグ操作がうまくできない利用者がある。ドラッグ操作にも代替手段を準備する必要がある。

悪い例

地図をピンチイン・ピンチアウトで拡大縮小できるが、その操作の代替操作がないため、タッチペンを利用した拡大縮小ができない。

良い例

地図をピンチイン・ピンチアウトで拡大縮小できるが、それとは別にプラスボタン・マイナスボタンが準備されており、簡単なジェスチャで拡大縮小ができる。

[12]操作のキャンセル

WCAG2.2 の達成基準

- 2.5.2 : レベル A ポインタのキャンセル

ルール

利用者が誤ってタッチ（マウスではクリック）した操作をキャンセルできるようにする。

具体的には以下の動作を実現する。

- 操作部分をタッチしただけでは動作せず、指を持ち上げた時に動作する。
- 利用者が誤った部分をタッチした時は、操作部分から指をずらすことでキャンセルできる。

ポイント

腕や手が不自由な利用者等、意図せず画面の操作部分にタッチしてしまう利用者にとって有用である。

[13]動きによる起動

WCAG2.2 の達成基準

・2.5.4：レベルA 動きによる起動

ルール

端末の動き(振る、傾ける、回転させる等)や利用者の動き(身振り・手振り・表情等)により呼び出せる機能を設定する場合、他の手段(1本指・タッチペンによるタッチ操作、キーボード等)の入力による起動に切り替えられるようにする。

その場合、誤って機能が呼び出されないよう、動きによる起動を無効化できるようにする。

ポイント

端末を振ることや傾けることが難しい利用者でも必要な機能を利用できるよう、代替手段を用意する必要がある。

悪い例

端末を振ることで呼び出せる共有機能が、その他のタップ・キーボード操作により実行できるように設定されていない。

良い例

端末を振ることで呼び出せる共有機能に他の実行用ボタンが用意されているほか、端末の動きを感知しないよう設定するためのボタンが設置されている。

[14]入力欄につける説明

WCAG2.2 の達成基準

- 2.4.6 : レベル AA 見出し及びラベル
- 2.5.3 : レベル A ラベルを含む名前(name)
- 3.3.2 : レベル A ラベル又は説明

ルール

テキストボックス、ラジオボタン、チェックボックス等入力フォームの部品には必ず、その用途や説明を記載し、入力フォームの関連付けを行う。

モバイルアプリのウェブ部分では、HTML の「for」属性で紐づけた「<label>」要素を提供する。

ポイント

音声入力で操作している利用者が、説明の文字を読み上げることで、的確に操作ができるようになる。

入力フォームの説明が文字で提示されていても、入力フォームと関連付けられていない場合、スクリーンリーダーの利用者にとって、入力や選択内容が分かりづらくなる。

HTML の title 属性や aria-label 属性のように画面上表示されない名前を部品に付ける場合は、画面上表示される名前と一致させることが望ましい。

✖ 悪い例

入力フォームの説明が記載されているが、「<label>」要素が使われておらず、またフォームと関連付けられていない。

```
<p>生年月日(半角8桁で入力)</p>  
<input type="text" >
```

○ 良い例

入力フォームの説明が「<label>」要素で記述されており、その内容が「for」属性により関連付けられている。

```
<label for="birthday" >生年月日(半角8桁で入力)</label>  
<input type="text" id="birthday" >
```

[15]入力欄のエラー回避・対応

WCAG2.2 の達成基準

- 3.3.1 : レベル A エラーの特定
- 3.3.2 : レベル A ラベル又は説明
- 3.3.3 : レベル AA エラー修正の提案
- 3.3.4 : レベル AA 誤り防止(法的、金融、データ)

ルール

入力フォームに入力条件（必須項目、全角のみ、半角英数のみ、文字数）等がある場合、その条件を明記する。条件は入力前に把握できるような位置に記載する。

必須項目が入力されていない、半角文字ではなく全角文字が入力されている等、自動的にエラーを検出した場合は、エラー箇所とエラー内容をテキスト情報で明示する。

個人情報の送信や、申請手続き等のために情報を送る際は、確認画面を準備する等、利用者が送信を取り消し、修正、確認できる画面を用意する。

ポイント

可能な限り入力エラーが起こらないように、入力の条件は分かりやすく記載すること。入力例を記載することも、誤りのない入力の助けとなる。

エラーが自動判別された際、エラー箇所を色のみで示すことがあるが、スクリーンリーダーの利用者は該当箇所を判別できない。また、エラー箇所が理解できても、修正方法が分からないことがある。自動的にエラーを検出する場合は、エラーの理由を明示して修正のヒントを提示する。

悪い例

- 未入力の必須項目があっても、色が変わるだけでエラーの箇所や原因が明記されていない。
- 半角で入力すべき項目を全角で入力したところ、エラーになるが、エラー理由が示されないため、修正方法が分からない。

○ 良い例

- 未入力の必須項目がある場合、その直前に「こちらは必須の項目です。」等、テキストによる通知が表示されている。
- 半角文字のみ受け付ける入力フォームに全角文字が入力されている場合、その直前に「電話番号は半角で入力してください」等と修正案が表示されている。

[16]入力補助

WCAG2.2 の達成基準

- 1.3.5 : レベル AA 入力目的の特定
- 3.3.7 : レベル A 冗長な入力項目

ルール

利用者に入力を求める場合、可能な限り自動入力機能を活用する。

利用者が同じ項目を複数回入力することなく、手続きを完了できるような仕組みを準備する。

ポイント

利用者の入力を極力減らすことで、入力に時間を要したり、入力間違いをしやすい利用者がスムーズに手続きを行えるようになる。

✕ 悪い例

- ウェブアプリの複数画面に配置されるテキストボックスにおいて、何度も同じ内容を入力する。

○ 良い例

- ウェブアプリのテキストボックスにおいて、HTML の autocomplete 属性が設定されており入力を補助している。
- 資料送付の申し込み画面において、申し込み者の氏名、住所、電話番号を入力した後、資料送付先が同じである場合、同じ情報をコピーするためのボタンが設置されている。

[17]画像などを用いた認証使用時の配慮

WCAG2.2 の達成基準

- 1.1.1：レベルA 非テキストコンテンツ

ルール

画像に描かれている文字を入力して認証する仕組等を設置する場合は、他の方法での認証方法を設置し、2つ以上の方法から利用者が選択できるようにする。

ポイント

不正なアクセスを防ぐために、視覚や聴覚を使って人間が操作していることを確認する仕組を設置していることがある。画像に描かれている文字を入力させる仕組や、指定された画像を選択する仕組等がその例である。それらは、視覚障害者にとってはバリアとなる。

悪い例

画像で描かれている文字を入力しないと次の画面に進めず、利用者登録ができない。

良い例

- 画像に描かれている文字を入力するか、音声を聞いて単語を入力するか選択できる。
- 複数の画像から指定された画像を選択する認証方法と、メールに送信されたワンタイムパスワードを入力する方法が選択できる。

[18]アクセシブルな認証

WCAG2.2 の達成基準

・3.3.8：レベル AA アクセシブルな認証（最低限）

ルール

ログイン等認証が必要な場合、以下のような認知や記憶に負担のないよう配慮を行う。

- ログイン ID やパスワードの入力部分でコピー&ペーストを禁止しない。
- パズルを解く、計算をする等の複雑な操作を利用者に求めない。

ポイント

認知・記憶に障害のある利用者の精神的負担を軽減することがこのルールの目的である。パスワードを記憶する、パズルを解く等は、認知・記憶に障害のある利用者の負担となる。

悪い例

メールで送付されたワンタイムパスワードを入力する仕組みであるが、入力エリアで文字の貼り付けを禁止している。

良い例

メールで認証用の URL を送付し、利用者がアクセスすることで認証を完了する仕組みになっている。利用者は入力等の負担がなく認証される。

[19]入力の時間制限

WCAG2.2 の達成基準

・2.2.1：レベルA タイミング調整可能

ルール

利用者に様々な情報の入力を求める場合、制限時間を設定しない。
やむを得ず制限時間を設定する場合は、制限時間を明示し、以下のいずれかの仕組みを準備する。

- 事前に解除する。
- 事前に制限時間を 10 倍以上に延長する。

ポイント

情報の入力に時間を要する利用者があり、時間制限がある場合、時間内に完了できないことがあるため、時間制限を設定しないことが推奨される。

悪い例

アンケートへの回答を入力する画面で 20 分の時間制限が設定されているにも関わらず、それについて通知されていない。

良い例

アンケートへの回答を入力する画面で、質問の前に制限時間に関する注意が記載されており、解除するボタンが設置されている。

[20]変化・追加されるテキストメッセージを通知する

WCAG2.2 の達成基準

- 4.1.2 : レベル A 名前(name)・役割(role)・値(value)
- 4.1.3 : レベル AA ステータスメッセージ

ルール

利用者の操作で変化した状態（チェックボックスやラジオボタンの選択等）について、利用者が使用する支援技術を通じて理解できるようにする。

また、モバイルアプリから通知される、操作の可否の結果、現在の状況、状況の変化のメッセージは、スクリーンリーダー等の支援技術の利用者にも伝わるように作成する。

※ WCAG2.2 では、マークアップ言語（HTML 等）を用いて作成しているものについて言及しているが、モバイルアプリにおいても可能な範囲で対応することが重要である。

ポイント

スクリーンリーダー等の利用者が認識できるよう、チェックボックスやラジオボタン、プルダウンメニュー等は、選択した内容が確認できるようにする。

また、スクリーンリーダーはフォーカスを当てている部分を読み上げることはできるが、モバイルアプリからの通知等はフォーカスが当たらない限り読み上げることができない。以下については、フォーカスが当たらなくても読み上げできるように設計することで利用者が現在の状況を知ることができる。

- ボタンを押すことで変化・追加されるテキスト
例：「〇〇件の検索結果を表示しました」「〇個のアイテムをカートに追加しました」
- アプリの処理中やダウンロード中を知らせるテキスト
例：「ダウンロード中〇〇%」
- 入力中に表示されるエラーメッセージ
例：「英数字のみを入力してください」

悪い例

- 入力フォームのチェックボックスにチェックを入れたが、チェックの有無が読み上げられず、スクリーンリーダーの利用者はチェックの有無を把握できない。
- 検索画面において、検索ワードを入力し検索ボタンを押すと、画面遷移を伴わずに検索結果が画面の一部を更新する形で挿入されるが、スクリーンリーダーに通知する仕組みがないため、スクリーンリーダーの利用者は画面の変化に気づくことができない。

良い例

- 入力フォームのチェックボックスにチェックを入れると、「チェック有り」とスクリーンリーダーが読み上げるため、チェックが入っていることを把握できる。
- 検索画面において、検索ワードを入力し検索ボタンを押すと、画面遷移を伴わずに検索結果が画面の一部を更新する形で挿入され、スクリーンリーダーは、「○件の検索結果が表示されました」と読み上げる。

[21]自動で追加表示されるコンテンツへの配慮(キーボード・マウス操作時)

WCAG2.2 の達成基準

- 1.4.13 : レベル AA ホバー又はフォーカスで表示されるコンテンツ

ルール

キーボード操作でボタンにフォーカスを当てる、マウス操作でポインタをボタンの上に移動しただけで、コンテンツが追加されないようにする。やむを得ず、表示されるコンテンツがある場合は、以下の配慮を行う。

- ポインタやフォーカスをその場から動かすことなく、表示されたコンテンツを非表示にする仕組（例：Esc キーを押す）を用意する。
- 追加されたコンテンツ部分にフォーカスやポインタを動かした際に、追加コンテンツが消えることなく操作・内容を確認できる。
- フォーカスやポインタを動かしても、自動的に追加コンテンツが消えないようにする。

ポイント

サブメニューの表示やヘルプメッセージの表示等、キーボードフォーカスやマウスポインタが当たっただけで表示されるコンテンツは、利用者にアクセシビリティ上の問題を起こすことがあるため、使用しないことが望ましい。

追加されたコンテンツを簡単な操作で消すことは、利用者が追加コンテンツを必要としていなかった場合に役立つ。

ただし、フォーカスやポインタを動かした際に追加コンテンツが消えてしまうと、キーボードを操作している利用者や画面を拡大している利用者が、追加コンテンツの内容を把握・操作できないことがある。そのため、フォーカスやポインタを動かしても追加コンテンツが消えないようにすることが大切である。

悪い例

「詳細」という文字にキーボードフォーカスを当てると詳しい説明が表示されるが、フォーカスを移動すると閉じてしまう。

メニューボタンにキーボードフォーカスを当てると、メニュー項目が表示されるが、キーボードで各項目を操作できない。

良い例

メニューボタンをフォーカスして各メニュー項目を表示させた後も、次の条件が満たされている。

- キーボードのESCキー押下で表示されたコンテンツを閉じることができる。
- 表示されたメニューをキーボードで操作できる。
- フォーカスをはずしても、自動的に追加コンテンツが消えない。

[22] 自動的な画面遷移を避ける

WCAG2.2 の達成基準

- 3.2.1 : レベル A フォーカス時
- 3.2.2 : レベル A 入力時

ルール

利用者が想定できないタイミングで、コンテンツの大きな変化を起こさない。

ポイント

画面が遷移する等コンテンツが大きく変化することを利用者が想定できていない場合、利用者がコンテンツを理解し、操作する上で支障が生じる。

悪い例

- キーボードで操作している利用者が、ボタンにフォーカスを当てたところ、別の画面に遷移した。
- 利用者が、ラジオボタンにチェックを入れただけで別の画面に遷移する。

良い例

利用者が、ラジオボタンにチェックを入れ、決定ボタンを押した後、画面が遷移する。

[23]操作ボタン、リンク設定のルール

WCAG2.2 の達成基準

- 1.1.1：レベルA 非テキストコンテンツ
- 2.4.4：レベルA リンクの目的(コンテキスト内)

ルール

操作ボタン、画面遷移するボタン等は、そのボタンの名前のみで、役割・遷移先の画面を把握できるように設定する。

ボタンに画像を用いる場合は、本ガイドライン[\[25\] 画像の代替テキスト](#)を参照し、テキスト情報を付加する。

- ※ モバイルアプリ内のウェブ部分で、リンクを設定する場合は、周囲のテキスト、直前の見出し等を組み合わせて、リンク先が理解できるようにする。

ポイント

ボタンを押下した後に何が起こるのか、どのような画面遷移があるのかを利用者が把握できるようにする。

前後の文章を読まなくても把握できることが望ましい。

悪い例

「〇〇の詳細はこちら」「△△の詳細はこちら」「□□の詳細はこちら」のように並んでいる。操作できる部分が「詳細はこちら」のみのため、スクリーンリーダーの利用者は操作部分を区別することができない。

良い例

「〇〇の詳細はこちら」全体が操作できるようになっており、内容だけでリンク先を理解できる。

[24]操作ボタンなどのサイズ

WCAG2.2 の達成基準

- 2.5.8 : レベル AA ターゲットのサイズ(最低限)

ルール

操作可能な部分は十分な大きさを確保し、確保できない場合は他の操作部分と十分な間隔をとること。

- 縦・横ともに 24×24 ピクセル以上を確保する。
- より小さいサイズの操作ボタンを使用する場合は、全ての操作ボタンを中心に 24 ピクセルの円を描いた時に、隣接する円同士が重ならないように配置する。

ポイント

高齢の利用者や腕や手先が不自由な利用者は、小さいボタンを押すことが難しく、誤って隣のボタンを操作してしまうことがある。

(4)非テキスト情報

[25]画像の代替テキスト

WCAG2.2 の達成基準

- 1.1.1：レベルA 非テキストコンテンツ

《意味のある画像》

ルール

意味のある画像(ページのコンテンツを伝える上で必要な画像)については、具体的な代替テキストを設定する。

ポイント

スクリーンリーダーの利用者は、目視で画像を確認することができないため、具体的な説明を代替テキストとして提供する。

✕ 悪い例

- 代替テキストが設定されていないため、スクリーンリーダーでは画像のファイル名等が読み上げられてしまう。
- 代替テキストが図表タイトルのみのため、その内容を把握することができない。

○ 良い例

- 適切な代替テキストが設定されているため、スクリーンリーダーで画像に描かれている内容を把握することができる。
- 画像の代替テキストとして、図表のタイトルとその内容を設定している。

《装飾画像》

ルール

ページの内容を伝える上で必要のない、装飾的な画像については代替テキストを設定しない。

ポイント

装飾画像の代替テキストとして、「装飾画像」等と設定してしまうと、不必要な情報が多くなり、スクリーンリーダーの利用者がコンテンツの内容を把握しづらくなる。

[26]音声の代替テキスト

WCAG2.2 の達成基準

- 1.2.1 : レベル A 音声のみ及び映像のみ(収録済)

ルール

音声ファイルを提供する場合、内容を書き起こしたテキストを合わせて掲載する。

ポイント

音声情報だけでは聴覚障害者は内容を把握することができない。

[27]映像の代替テキスト

WCAG2.2 の達成基準

- 1.2.1 : レベル A 音声のみ及び映像のみ(収録済)

ルール

音声のない動画を表示する場合、内容を書き起こしたテキストを合わせて掲載する。

ポイント

映像情報だけでは視覚障害者は内容を把握することができない。

[28]字幕・キャプション

WCAG2.2 の達成基準

- 1.2.2 : レベル A キャプション(収録済)
- 1.2.4 : レベル AA キャプション(ライブ)

ルール

映像と音声の両方を含む動画では、キャプション(字幕)を提供しなければならない。

ポイント

聴覚障害者でも音声情報を識別できるように、話している登場人物、会話の内容、意味のある効果音等を説明したキャプションを提供する。

悪い例

音声付きの動画が掲載されているが、字幕が提供されていない。

良い例

音声付きの動画に、必要な音声情報を全て説明した字幕が提供されている。

[29]動画の音声解説

WCAG2.2 の達成基準

- 1.2.3 : レベル A 音声解説、又はメディアに対する代替(収録済)
- 1.2.5 : レベル AA 音声解説(収録済)

ルール

映像と音声の両方を含む動画を提供する際は、シーンの切り替わり・登場人物・キャプションの文字情報等、内容を理解するために必要な情報を説明する音声解説を提供する。

代替手段として、音声解説の提供が難しい場合は、内容を説明する台本のような内容の文章を提供することを検討すること。ただし、この方法ではレベル AA を満たすことはできない。

ポイント

映像情報を視認できない利用者でも必要な情報を得られるよう、音声による解説を提供する。音声解説は、セリフとセリフの間等、本来の音声に重ならないようにする。

悪い例

動画が掲載されているが、映像を理解するための解説が提供されていない。

良い例

通常の動画とは別に音声解説版の動画が提供されている。

(5)色及び形

[30]色や形など感覚的な特徴だけで情報を区別しない

WCAG2.2 の達成基準

- 1.3.3 : レベル A 感覚的な特徴
- 1.4.1 : レベル A 色の使用

ルール

閲覧・操作に必要な情報は、形、色、大きさ、方向、音等の感覚的な特徴だけに依存して提供してはいけない。

ポイント

視覚障害者は色・形・大きさ等を認識できないため、感覚的な特徴のみの情報だけでは内容を把握することができない。文字情報等の正しく区別できる情報を提供する。
聴覚障害者は音を認識できないため、視覚的に分かる情報を提供する。

悪い例

- 「次に進むためには丸いボタンを押してください」と形のみで次の操作の手順を説明している。
- 写真のアップロード完了が音のみで知らされている。
- 人名の一覧があり、「青色は新しい役員です」と色のみで情報を区別している。
- 棒グラフの各項目が、色のみで塗り分けられているため、色覚特性によっては色の判別がでない。

良い例

- 「次に進むためには『確定』と書かれた丸いボタンを押してください」と形と文字で次の操作の手順を説明している。
- 写真のアップロード完了時、音とともに「アップロード完了」という文字で知らされている。
- 人名の一覧があり、「青色は新しい役員です」と色で情報を区別し、人名の後ろには「(新役員)」と文字も書かれている。
- 棒グラフの各項目が、色と模様の両方で区別できるように説明している。

[31]文字色と背景色の組み合わせ

WCAG2.2 の達成基準

- 1.4.3 : レベル AA コントラスト(最低限)

ルール

テキストや画像に含まれる文字は、文字色と背景色の間に少なくとも 4.5 : 1 以上のコントラスト比を確保しなければならない。

- ※ 操作できないボタン、装飾目的のもの、視認できないもの、ロゴやブランドデザインの一部になっているものは対象外とする。

ポイント

中程度の弱視の利用者や高齢の利用者が、特別な設定なく情報を得られるようにするための配慮。この配慮は色を識別しづらい色覚特性のある利用者にも役立つ。

- ※ コントラスト比の計算は複雑であるため、コントラスト比をチェックできるツールの利用を推奨する。

悪い例

白色の背景に赤色の文字が書かれており、コントラスト比が 4.1 : 1 になっている。

良い例

画像で白色の背景に赤色の文字が書かれているが、暗い色で縁取りされており、コントラスト比が 5 : 1 になっている。

[32]文字以外の意味のあるコンテンツと周囲の色の組み合わせ

WCAG2.2 の達成基準

- 1.4.11 : レベル AA 非テキストのコントラスト

ルール

操作ボタン、意味のあるアイコン、画像内のグラフ等、閲覧・操作するために理解が必要なものは、周囲と少なくとも 3 : 1 のコントラスト比を確保しなければならない。

ポイント

中程度の弱視の利用者や高齢の利用者が、特別な設定なく情報を得られるようにするための配慮。この配慮は色を識別しづらい色覚特性のある利用者にも役立つ。文字と背景色の組み合わせのコントラスト比よりは低い値が良い。

- ※ このコントラスト比の計算は複雑であるため、コントラスト比をチェックできるツールの利用を推奨する。

(6)文字

[33]画像化された文字の使用制限

WCAG2.2 の達成基準

・1.4.5 : レベル AA 文字画像

ルール

シンプルなフォントを使った文字は画像化せず、テキストで記載する。

※ モバイルアプリ内で HTML を使用している部分では、CSS で装飾可能な範囲のものはテキスト化する。

ポイント

文字を画像化した場合、ハイコントラストモードにしても色が切り替わらなかったり、画面を拡大すると画像が粗くなってしまいう等の問題から、弱視者が必要な情報を識別できなくなってしまう可能性が高いため、できる限りテキストで表現する。

✖ 悪い例

- 「新着情報」というシンプルなテキストが画像化されている。
- 講演の内容について、写真部分とテキスト部分の1枚の画像で提供されている。(テキスト部分は装飾がない通常のテキストで表現できるデザインである。)

○ 良い例

- 「新着情報」という文字が、画像ではなくテキストで表示されている。
- 講演の内容は、写真とテキストを分けて記載している。

[34]位置調整目的のスペースの禁止

WCAG2.2 の達成基準

- 1.3.2 : レベル A 意味のあるシーケンス

ルール

単語の途中にスペースや改行を入れない。

ポイント

単語の途中にスペースや改行があると、スクリーンリーダーで正しく読み上げられない。

見た目を整えたい場合は、文字間隔を設定する仕組みを活用する。

HTML では、CSS を利用し、調整することができる。

悪い例

単語内にスペースを入れている。

- 電話 → 「デン ハナシ」 と読み上げられることがある。
- 住所 → 「ジュウ トコロ」 と読み上げられることがある。

良い例

どの単語内にも見た目を整えるためのスペースが入っておらず、スクリーンリーダーで正しく読み上げられる。

[35]テキストの間隔

WCAG2.2 の達成基準

・1.4.12：レベルAA テキストの間隔

ルール

指定できる場合は、行間、段落間、文字間、単語間のスペースを適切にとり、また変更できるようにする。

以下を基準として求める。

- 行の間隔（行送り）を少なくともフォントサイズの1.5倍以上に設定
- 段落後の間隔を少なくともフォントサイズの2倍以上に設定
- 文字の間隔（字送り）を少なくともフォントサイズの0.12倍以上に設定
- 単語の間隔を少なくともフォントサイズの0.16倍以上に設定

ポイント

利用者によっては、テキストの読みやすさを向上させるためにテキストの間隔やフォントの種類を変更することがある。このルールはそのようなカスタマイズの幅を広げるためのものである。

ただし、ネイティブアプリでは、利用者によるカスタマイズは難しいことが想定されるため、予めこの基準に沿ってデザインすることが求められる。

(7)音

[36]音を使用する際の注意

WCAG2.2 の達成基準

- ・1.4.2 : レベルA 音声の制御

ルール

モバイルアプリの起動時や画面切り替え時等、利用者が意図しない状況で音声自動再生されないようにする。

ポイント

自動再生される音声は、スクリーンリーダーの音声を聞きながら操作している利用者にとって、操作の妨げになることがある。そのため、音声は自動再生させないようにする。

動画閲覧のモバイルアプリで音声再生が必要な場合は、再生ボタンを設置する等、利用者の操作により再生されるようにする。

悪い例

モバイルアプリを起動すると BGM が自動的に再生され、それを停止するためのボタン等が一切用意されていない。

良い例

モバイルアプリを起動しても音声自動的に再生されることはなく、「音楽を再生」というボタンを押すと初めて BGM が流れるようになる。

(8)速度

[37]変化・移動する画像や文字の禁止

WCAG2.2 の達成基準

- 2.2.1 : レベルA タイミング調整可能
- 2.2.2 : レベルA 一時停止、停止、非表示

ルール

動きのあるコンテンツは、利用者が読む時間を十分に確保できるように、以下の仕組を用意する。

- 表示後、5秒以内に停止する。
 - 一時停止または停止、非表示にする仕組を準備する。
- ※ 動きのあるコンテンツとは、切り替わる画像、スクロールするテキスト、点滅する箇所等を指す。

ポイント

利用者の中には、文字を読むことに多くの時間を必要とする人がいる。

コンテンツの動きを止めることは、動きのあるコンテンツがあると集中できず、他の部分を読むことができない人にとっても有用である。

[38]激しい画面点滅の禁止

WCAG2.2 の達成基準

- 2.3.1 : レベルA 3回の閃光、又は閾値以下

ルール

画面の広範囲が点滅しないようにする。

ポイント

激しい画面の点滅は、利用者が光感受性による発作を引き起こす可能性がある。基準を満たす条件は様々あるが、利用者の安全を考慮し、広範囲の点滅を禁止する。

(9)言語

[39]言語の指定

WCAG2.2 の達成基準

- 3.1.1 : レベル A ページの言語
- 3.1.2 : レベル AA 一部分の言語

ルール

表示しているコンテンツの言語を指定できる場合は、明示する。
画面全体と異なる言語が使われている部分がある場合、段落や単語ごと等、細かく指定できる場合は、明示はする。

ポイント

言語を明示することで、スクリーンリーダー等が正しい発音で読み上げることができる可能性がある。特にアルファベットで表記する言語では、同じスペルでも異なる読み方をすることがあるため有用である。

ウェブアプリ、ハイブリッドアプリのウェブ部分ではHTMLの lang 属性を使用できる。