

2024-01～ WCAG解説書 輪読会

対象	https://waic.jp/translations/WCAG21/Understanding/
原文	https://www.w3.org/WAI/WCAG21/Understanding/ https://www.w3.org/WAI/WCAG22/Understanding/ （日本語訳はまだ）

WCAG解説書輪読会2024

やるぞお～

期間	2024年1月22日～
頻度	2週毎

WCAG解説書とは？

WCAG = Web Content Accessibility Guidelinesはウェブアクセシビリティに関する国際標準です。

WCAG解説書は、WCAGを理解しやすく解説するための文書です。ガイドラインの背後にある思想、特定の要件、およびそれらを実装する方法についての詳細な情報を提供します。

[WCAG 2.1 解説書](#)

なんでWCAG解説書？

とても学びがあるからです。

アクセシビリティはなかなか捉えどころのない概念です。その要因の一つには、アクセシビリティで恩恵を受けるユーザーや状況というものが非常に多岐にわたっていて、「こういう人のためのものだ」と一言で言い切れないことが挙げられると思います。

WCAG解説書は、WCAGの達成基準がなぜそのようになっているのかの意図や背景、経緯を丁寧に説明してくれています。ときには具体的な障害や状況の例をふまえて説明されるので、理解が深まりますし、解説書の通読によってアクセシビリティが達成したいことの全体像がぼんやりと見えてくるのではないかと考えました。

どうやってやる？

輪番制とする。

当番になった者は、割り当てられた担当範囲を事前に通読し、自分なりにConfluenceなどにまとめてくる。

以下のような観点をもって見てみると知識リンクが広がっていいかもしれない。

- LIFULLやLIFULL HOME'Sで好事例や失敗例はあるか
 - 他のウェブサイトや現実世界での事例があるか
- [LIFULLアクセシビリティガイドライン](#)ではどうなっているか
- 単純な感想

担当範囲

回	日付	担当範囲	担当者
1	2024年1月22日	<ul style="list-style-type: none">● WCAG 2.1 解説書のイントロダクション● WCAG 達成基準の達成方法を理解する	鳶田喬行
2	2024年2月5日	<ul style="list-style-type: none">● ガイドライン 1.1: テキストによる代替を理解する<ul style="list-style-type: none">○ 達成基準 1.1.1: 非テキストコンテンツを理解する	嶋岡勝仁
3	2024年3月4日	<ul style="list-style-type: none">● ガイドライン 1.2 時間依存メディア<ul style="list-style-type: none">○ https://waic.jp/translations/WCAG21/#time-based-media○ Web Content Accessibility Guidelines (WCAG) 2.1 (日本語訳)	鈴木亜依
4	2024年4月1日	<ul style="list-style-type: none">● ガイドライン 1.3 適応可能	千葉穂乃美
5	2024年5月16日	<ul style="list-style-type: none">● 1.3.3: 感覚的な特徴を理解する● https://waic.jp/translations/WCAG21/Understanding/sensory-characteristics.html● 十分な達成方法<ul style="list-style-type: none">○ 理解するために感覚的な情報のみに依存するような項目に対して、テキストによる識別を別の方法で提供する● 失敗例<ul style="list-style-type: none">○ 形状又は位置のみでコンテンツを特定している○ 情報を伝えるために、グラフィカルなシンボルのみを使用している	久保田大輔
6	2024年5月30日	<ul style="list-style-type: none">● 1.3.4 表示の向き● 1.3.5 入力目的の特定● 1.3.6 目的の特定	Takuya Nakajima
7	2024年6月27日	<ul style="list-style-type: none">● ガイドライン 1.4 判別可能	鳶田喬行

		<ul style="list-style-type: none"> 1.4.1 色の使用 1.4.2 音声の制御 1.4.3 コントラスト (最低限) 	
8	2024年7月11日 → 2024年8月22日	<ul style="list-style-type: none"> 1.4.4 テキストのサイズ変更 1.4.5 文字画像 1.4.6 コントラスト (高度) 1.4.7 小さな背景音、又は背景音なし 	鈴木亜依 → 嶋岡勝仁
9	2024年10月17日	<ul style="list-style-type: none"> 1.4.6 コントラスト (高度) 1.4.7 小さな背景音、又は背景音なし 	嶋岡勝仁
10	2024年11月14日	<ul style="list-style-type: none"> 達成基準 1.4.8 視覚的提示 達成基準 1.4.9 文字画像 (例外なし) 達成基準 1.4.10 リフロー <p>余裕があれば</p>	千葉穂乃美
11		<ul style="list-style-type: none"> 	久保田大輔
12	2024年11月28日	<ul style="list-style-type: none"> 達成基準 1.4.11 非テキストのコントラスト 達成基準 1.4.12 テキストの間隔 達成基準 1.4.13 ホバー又はフォーカスで表示されるコンテンツ 	Takuya Nakajima
13	2025年1月9日	<ul style="list-style-type: none"> 達成基準 2.1.1 キーボードを理解する 達成基準 2.1.2 キーボードトラップなしを理解する 	鈴木亜依
14	2025年1月23日	<ul style="list-style-type: none"> 達成基準 2.1.3: キーボード (例外なし)を理解する 達成基準 2.1.4: 文字キーのショートカットを理解する 達成基準 2.2.1: タイミング調整可能を理解する 	千葉穂乃美
15	2025年2月6日	<ul style="list-style-type: none"> 達成基準 2.2.2: 一時停止、停止、非表示を理解する 達成基準 2.2.3: タイミング非依存を理解する 達成基準 2.2.4: 割り込みを理解する 	Takuya Nakajima
16	2025年3月6日	<ul style="list-style-type: none"> 達成基準 2.2.5 達成基準 2.2.6 達成基準 2.3 	嶋岡勝仁
	2025年4月3日	<ul style="list-style-type: none"> 達成基準 2.3.1 3回の閃光、又は閾値以下を理解する 達成基準 2.3.2 3回の閃光を理解する 達成基準 2.3.3 インタラクションによるアニメーションを理解する 	秦那実
	2025年4月17日	<ul style="list-style-type: none"> 達成基準 2.4 ナビゲーション可能 達成基準 2.4.1 ブロックスキップ 達成基準 2.4.2 ページタイトル 	千葉穂乃美
	2025年5月15日	<ul style="list-style-type: none"> 達成基準 2.4.3 フォーカス順序 	鈴木亜依

	2025年6月26日	<ul style="list-style-type: none"> ● 達成基準 2.4.4 リンクの目的 (コンテキスト内) ● 達成基準 2.4.5 複数の手段 ● 達成基準 2.4.6 見出し及びラベル ● 達成基準 2.4.7 フォーカスの可視化 	Takuya Nakajima
	2025年8月7日	<ul style="list-style-type: none"> ● 2.4.8 現在位置 ● 2.4.9: リンクの目的 	秦那実
	2025年8月21日	<ul style="list-style-type: none"> ● 達成基準 2.4.10 セクション見出し ● 達成基準 2.4.11 隠されないフォーカス (最低限) ● 達成基準 2.4.12 隠されないフォーカス (高度) ● 達成基準 2.4.13 フォーカスの外観 	千葉穂乃美
	2025年9月4日	<ul style="list-style-type: none"> ● 達成基準 2.5.1 ポインタのジェスチャ ● 達成基準 2.5.2 ポインタのキャンセル ● 	Takuya Nakajima
	2025年9月19日	<ul style="list-style-type: none"> ● 達成基準 2.5.3 ラベルを含む名前 (name) ● 達成基準 2.5.4 動きによる起動 	Takuya Nakajima
	2025年10月2日	<ul style="list-style-type: none"> ● 達成基準 2.5.5 ターゲットのサイズ (高度) 	Takuya Nakajima
	2025年10月16日	<ul style="list-style-type: none"> ● 達成基準 2.5.6 入力メカニズムの共存 ● 達成基準 2.5.7 ドラッグ操作 	秦那実
	2025年11月27日	<ul style="list-style-type: none"> ● 達成基準 2.5.8: ターゲットのサイズ (最低限) 	千葉穂乃美
	2025年12月11日	<ul style="list-style-type: none"> ● 3.1.1 ページの言語 ● 3.1.2 一部分の言語 ● 3.1.3 一般的ではない用語 	Takuya Nakajima
	2026年1月8日	<ul style="list-style-type: none"> ● 3.1.4 略語 ● 3.1.5 読解レベル 	宮崎陽喜

2024年1月22日

WCAG 2.1 解説書のイントロダクション

- WCAG 2.1 解説書の目的
 - **WCAG 2.1** 解説書は、"**Web Content Accessibility Guidelines 2.1**"を理解して実践するために不可欠な案内書である。
 - 達成基準の意図や、どのようにガイドラインと達成基準が連携しているのかの解説
 - 達成基準を満たすのに十分な達成方法や、達成方法の組合せの事例の紹介

解説書の読み方

- 解説書の構成
 - ガイドラインごとに構成されている
 - 「X.X {ガイドライン}」
 - ガイドラインの意図
 - ガイドラインに関係あるが、そのガイドラインの達成基準のいずれにも特に関係のない参考達成方法
 - 「X.X.X {達成基準}」
 - WCAG 2.1 に書かれている達成基準
 - その達成基準の意図
 - メリット (その達成基準が、どのように障害のある利用者の役に立つのか)
 - 事例
 - 関連リソース
 - その達成基準の十分な達成方法及び達成方法の組合せ
 - この達成基準のよくある失敗例
 - その達成基準を満たすための要求を超えているが、一部のあるいはすべてのコンテンツをさらにアクセシブルにするために用いることのできる参考達成方法。参考達成方法を使用することで、宣言する適合レベルに影響を及ぼすことはない。
 - この達成基準における重要な用語 (WCAG 2.1 の用語集から引用)
- 解説書へのアクセス方法

- WCAGの達成基準からリンクされている

§ **Guideline 1.1 Text Alternatives**

Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.

[Understanding Text Alternatives](#)
[How to Meet Text Alternatives](#)

§ **Success Criterion 1.1.1 Non-text Content**

(Level A)

All **non-text content** that is presented to the user has a **text alternative** that serves the equivalent purpose, except for the situations listed below.

Controls, Input

If non-text content is a control or accepts user input, then it has a **name** that describes its purpose. (Refer to [Success Criterion 4.1.2](#) for additional requirements for controls and content

アクセシビリティの四つの原則

1. 知覚可能
 - 利用者が提示されている情報を知覚できなければならない
 - 利用者の感覚すべてに対して知覚できないものであってはならない
2. 操作可能
 - 利用者がインタフェースを操作できなければならないこと
 - 利用者の実行できないインタラクションを要求してはならない
3. 理解可能
 - 利用者がユーザインタフェースの操作と情報とを理解できなければならない
 - コンテンツ又は操作が、理解できないものであってはならない
4. 堅牢
 - 利用者が技術の進歩に応じてコンテンツにアクセスできなければならない
 - 技術やユーザエージェントが進化していったとしても、コンテンツはアクセシブルなままであるべき

ガイダンスのレイヤー

- ガイドライン
 - ガイドラインの目的
 - コンテンツが、できるだけ多くの利用者にとってそのままアクセシブルであるようにすること
 - 様々な利用者の感覚的、身体的及び認知的能力に合った様々な形態で再提示できるようにすること
- 達成基準
 - WCAG 2.1 に適合するためには何をしなければならないのかを具体的に記述したもの
 - それぞれの達成基準は、特定のウェブコンテンツをその達成基準によってテストしたときに適合もしくは不適合が判別できるように記述されている

- ウェブコンテンツ技術に依存しないように書かれている
- コンテンツが達成基準を満たしていたとしても、そのコンテンツは様々な障害のある利用者にとって利用可能であるとは限らない
 - ある利用者にとってのアクセシビリティを確保する上では、定性的なヒューリスティック評価が重要
 - ユーザビリティテストが推奨される
- 様々な種類の障害のある人がどのようにウェブを利用しているのかを理解している人が、コンテンツをテストすべき
- 十分な達成方法及び参考達成方法、及び失敗例
 - 次節

WCAG 達成基準の達成方法を理解する

特定のウェブ技術における個別のガイダンス。

コード例、リソース及びテストを含む

<https://waic.jp/translations/WCAG21/Techniques/>

WCAG 2.1 のガイドライン及び達成基準は、動的アプリケーション、モバイル、デジタルテレビなどを含む現在及び将来のウェブ技術に対して広く適用可能となるよう設計されている。これらは安定しており変わることはない。

達成方法は、参考情報である。

達成方法は必須要件ではない。

【注意】

WCAG 2.1 への適合を判断する根拠として達成方法は使えない。

- 十分な達成方法
 - 達成基準を満たすのに信頼できる方法
 - 十分な達成方法を正しく用い、かつそれが利用者に対してアクセシビリティ サポートドならば、その達成基準を満たしている
 - アクセシビリティ サポートド: その技術がユーザエージェント及び支援技術によって利用者が使用可能であること。
 - 十分な達成方法以外にも、達成基準を満たす方法はある
 - 「かつ」と書いてある達成方法は全部達成しないとだめ
- 参考達成方法
 - アクセシビリティを改善するために推奨される方法
 - さまざまな理由で、十分な達成方法とはならない:
 - 達成基準のすべての要求事項を満たすのに十分でない。
 - まだ安定していない技術に基づいている。
 - 多くの場合においてアクセシビリティ サポートドでない (例えば、支援技術が達成方法に対応できていない)。

- テスト可能でない。
 - ある状況においてその達成方法が適用可能でないあるいは実用的でなく、かつその他の利用者に対してはアクセシビリティを向上させる一方、一部の利用者に対してアクセシビリティを低下させる可能性がある。
 - 達成基準そのものには対処しておらず、代わりに関連するアクセシビリティ上の利点を提供している。
- 失敗例
 - アクセシビリティ上のバリアを引き起こし、特定の達成基準への不適合を招くもの

2024年2月5日

ガイドライン 1.1: テキストによる代替を理解する

- 1.xは全てアクセシビリティ四原則の「知覚可能」に該当
- すべての非テキストコンテンツには、大活字、点字、音声、シンボル、平易な言葉などの利用者が必要とする形式に変換できるように、テキストによる代替を提供すること。
 - もうこれに全て集約されてるという感じだけど、全ての非テキストコンテンツにはテキストによる代替を提供するべし
 - ただし以下は除く
 - コントロール、入力
 - 非テキストコンテンツが、コントロール又は利用者の入力を受け付けるものであるとき、その目的を説明する[名前 \(name\)](#)を提供している。(コントロール及び利用者の入力を受け入れるコンテンツに関するその他の要件は、[Success Criterion 4.1.2](#)を参照。)
 - フォームとか。nameやroleが指定されていた場合、そのコンテンツの役割は明らかであるため、テキストでさらに代替する必要はないってイメージ。
 - 時間依存メディア
 - 非テキストコンテンツが、時間に依存したメディアであるとき、テキストによる代替は、少なくとも、その非テキストコンテンツを識別できる説明を提供している。(メディアに関するその他の要件は、[ガイドライン 1.2](#)を参照。)
 - 音声とか映像とか。これは次回やる部分。
 - テスト
 - 非テキストコンテンツが、[テキスト](#)で提示されると無効になるテスト又は演習のとき、テキストによる代替は、少なくともその非テキストコンテンツを識別できる説明を提供している。
 - テキストで代替してしまうとテストにならないやつのこと？

- シマウマの画像を出して、これは何の画像ですか？っていうのを聞きたいときに、代替テキストで「シマウマ」ってやると答えがバレちゃうからの？
- 感覚的
 - 非テキストコンテンツが、**特定の感覚的体験**を創り出すことを主に意図しているとき、テキストによる代替は、少なくともその非テキストコンテンツを識別できる説明を提供している。
 - 例)フルートのソロ演奏、視覚芸術の作品など
- CAPTCHA
 - 非テキストコンテンツが、コンピュータではなく人間がコンテンツにアクセスしていることを確認する目的で用いられているとき、テキストによる代替は、その非テキストコンテンツの目的を特定し、説明している。なおかつ、他の感覚による知覚に対応して出力する CAPTCHA の代替形式を提供することで、様々な障害に対応している。
 - よくあるテキスト入力させるやつ。
- 装飾、整形、非表示
 - 非テキストコンテンツが、**純粋な装飾**である、見た目の整形のためだけに用いられている、又は利用者に提供されるものではないとき、**支援技術**が無視できるように実装されている。
 - spacer.gifとか（古い！）

2024年3月18日

ガイドライン 1.2 時間依存メディア

時間依存メディアには代替コンテンツを提供すること。

言葉説明的なやつ

メディアによるテキストの代替	<p>テキストで (直接又はテキストによる代替によって) 既に提示されている情報以上のものを提示していないメディア。メディアによるテキストの代替は、テキストを代替する提示の恩恵を受ける人たちのために提供される。</p> <p>テキストの代替メディアになりうるのは、音声しか含まないメディア、映像しか含まない (手話の映像を含む) メディア、又は音声付映像メディアである。</p>
時間依存メディアに対する代替コンテンツ:	時間依存の視覚的及び聴覚的情報を正しい順序で説明したテキストを含み、あらゆる時間依存のインタラクションによる結果を得る手段を提供している文書。
同期したメディア:	<p>情報を提示するために、他のフォーマットと同期した 音声 もしくは 映像、及び／又は時間に依存するインタラクティブな構成要素と同期した音声もしくは映像。</p> <p>ただし、そのメディアが メディアによるテキストの代替 であって、そのように明確にラベル付けされているものは除く。</p>

(レベル A)

●達成基準 1.2.1 音声のみ及び映像のみ (収録済)

収録済の **音声** しか含まないメディア及び収録済の **映像** しか含まないメディアは、**時間依存メディアに対する代替** コンテンツによってコンテンツと同等の情報を提供している。(レベル A)

●達成基準 1.2.2 キャプション (収録済)(レベル A)

同期したメディア に含まれているすべての **収録済の音声** コンテンツに対して、**キャプション** が提供されている。ただし、その同期したメディアが **メディアによるテキストの代替** であって、メディアによる代替であることが明確にラベル付けされている場合は除く。

●達成基準 1.2.3 音声解説、又はメディアに対する代替 (収録済)(レベル A)

同期したメディア に含まれている **収録済の映像** コンテンツに対して、**時間依存メディアに対する代替** コンテンツ又は **音声解説** が提供されている。ただし、その同期したメディアが **メディアによるテキストの代替** であって、メディアによる代替であることが明確にラベル付けされている場合は除く。

(レベル AA)

- 達成基準 1.2.4 キャプション (ライブ)

同期したメディアに含まれているすべてのライブの音声コンテンツに対してキャプションが提供されている。

- 達成基準 1.2.5 音声解説 (収録済)

同期したメディアに含まれているすべての収録済の映像コンテンツに対して、音声解説が提供されている。

(レベル AAA)

- 達成基準 1.2.6 手話 (収録済)

同期したメディアに含まれているすべての収録済の音声コンテンツに対して、手話通訳が提供されている。

- 達成基準 1.2.7 拡張音声解説 (収録済)(レベル AAA)

前景音の合間が、音声解説で映像の意味を伝達するのに不十分な場合、同期したメディアに含まれているすべての収録済の映像コンテンツに対して、拡張音声解説が提供されている。

- 達成基準 1.2.8 メディアに対する代替 (収録済)(レベル AAA)

すべての収録済の同期したメディア及びすべての収録済の映像しか含まないメディアに対して、時間依存メディアに対する代替コンテンツが提供されている。

- 達成基準 1.2.9 音声のみ (ライブ)(レベル AAA)

ライブの音声しか含まないコンテンツに対して、それと同等の情報を提示する、時間依存メディアの代替コンテンツが提供されている。

2024年4月1日

[ガイドライン 1.3 適応可能](#)

<https://jira.next-group.jp/wiki/x/wMyRG>

次回

<https://www.w3.org/WAI/WCAG21/Understanding/sensory-characteristics>
<https://waic.jp/translations/WCAG21/Understanding/sensory-characteristics.html>

2024年5月30日

達成基準 1.3.4: 表示の向きを理解する

<https://waic.jp/translations/WCAG21/Understanding/orientation.html>

原文:コンテンツは、その表示及び操作を、縦向き (portrait) 又は横向き (landscape) などの単一の向きに制限しない。ただし、その表示の向きが必要不可欠な場合は例外とする。

ざっくりいうと...

コンテンツの表示の向き(**orientation**)を制限しちゃいけないって話
ここで言ってる向きはスマホ回転させたときに横表示とか縦表示になるあれに対応しようね(縦表示限定コンテンツとかにしちゃダメよってこと)

なんで...?

利用者の中には、(例えば電動車椅子の腕に) 固定された向きでデバイスを取り付けている人もいるよね

ロービジョンの利用者とかで横長のコンテンツを見るときにテキストサイズを大きくしたい時とかに回転を利用して解消しようとするケースとかに対応できる

例外は...?

向きを限定することが不可避なやつ。たとえばピアノの鍵盤アプリとか。

対応事例

- アマプラの動画とか
- Kindleとか

達成基準 1.3.5: 入力目的の特定を理解する

<https://waic.jp/translations/WCAG21/Understanding/identify-input-purpose.html>

原文:

達成基準 1.3.5 入力目的の特定 (レベル AA): 利用者の情報を集める入力フィールドのそれぞれの目的は、次の場合にプログラムによる解釈が可能である:

- 入力フィールドが、ユーザインタフェース コンポーネントの入力目的の節で示される目的を提供している、かつ
- フォーム入力データとして想定される意味の特定をサポートする技術を用いて、コンテンツが実装されている。

ざっくりいうと...

フォームの入力フィールドが何のためのものかをプログラムが理解できるようにするルール。

なんで...?

ユーザーがどんな情報を入力すればいいかを簡単に理解できるようにするため。

ブラウザや支援技術がそれみて、オートコンプリートとか提供してくれるようになれば、使いやすさが大幅にアップするよね、これは特に認知障害もってる人には大助かり

運動障害もっててうまく入力するのが難しい人もオートコンプリートがあると楽になるねえ

他にも文字読むのが苦手な人のために支援技術がそれみて見慣れたアイコン添えてあげるとかいんな用途につかえるネ

どうしたら実現できるの？

Webの場合はフォームコントロールにtype属性つけるだけでなくautocomplete属性もつけてあげるときっちり明示できていいね。

type属性だけだと値がtextの時とかどうしてもざっくりしちゃうからなだけつけてあげるといいね

<https://developer.mozilla.org/ja/docs/Web/HTML/Attributes/autocomplete>

達成基準 1.3.6: 目的の特定を理解する

<https://waic.jp/translations/WCAG21/Understanding/identify-purpose.html>

原文:

達成基準 1.3.6 目的の特定 (レベル AAA): マークアップ言語で実装されたコンテンツでは、ユーザインタフェース コンポーネント、アイコン、領域の目的はプログラムによる解釈が可能である。

ざっくりいうと...

マークアップ言語で作成されたコンテンツのアイコンや領域などの目的を、プログラムが理解できるようにするルール

ここで言っているのは、要素の使い道をわかりやすくするためのセマンティクスやメタデータを追加しよう
ねってこと

1.3.5は入力フィールドに注目してたけど今回はコンテンツ全般にフォーカスしてるネ

なんで...?

きちんとマークアップして要素の意味を明示しておけば、ブラウザや支援技術がそれがなんなのかを
あまり語彙をもっていない人に使い慣れたシンボル(アイコン)で教えてくれたりできるね

領域や重要なコンテンツもきっちりと明示しておけば重要じゃない領域を削除したり、重要なところを
強調表示したり、ブラウザや支援技術側もいろいろやりようがあるネ
これは記憶や集中力・注意力に弱みのあるユーザーにとっても優しい支援になるね
他にもランドマークジャンプとか便利よね

脱線するけど...

AAC(補助代替コミュニケーション)って知ってる?

<https://tokyo-ohc.org/wp/wp-content/uploads/2017/03/s12.pdf>

W3CとBCIが共同公開したAACのシンボル

<https://www.w3.org/TR/aac-registry/>

どうしたら実現できるの?

領域の明示に関してはランドマークroleきちんとつけて画面領域を表現しようネ

要素の意図を表現するためには最近まで作業勧められてたpersonalization系の仕様があったので
当初はそれを指してたっばい
5/2に一部アーカイブされたけどもともとあった aria-impotanceとかcoga-simplification とかがそれに
あたる

<https://www.w3.org/TR/2017/WD-personalization-semantics-1.0-20170606/#coga-simplification>

現在はadaptって仕様になって策定が進められてる

<https://www.w3.org/TR/adapt/>

- Symbol
- Help and Support
- Tools

の3モジュールを公開予定

いずれもadapt-*属性が提供される予定になっていてSymbolモジュールがシンボルコミュニケーショ
ン用

```
<span adapt-symbol="13621">Cup</span> of <span  
adapt-symbol="17511">Tea</span>
```

みたいなノリ

help and supportは

```
It is <span adapt-literal="raining hard">raining cats and dogs</span>
```

みたいなノリ(raining cats and dogsは土砂降りって意味の慣用句) adapt-literalは比喩の意味を伝えるもので他にも言い換えのためのadapt-alternativeとかヘルプテキストの種類を指定するadapt-helptypeとかいろいろある

toolsはちょっとごった煮感あるけど

```
<div role="alert"
  adapt-messageimportance="medium"
  adapt-messagefrom="my calender"
  adapt-messagecontext="work"
  adapt-messagetime="10.02.2017.00.00-16.02.2017.00.00">
  Renew your driving license this week
</div>
```

みたいな重要性や補助情報を提供してたりする
他にもパンくずの完了状態を示したりとか

```
<div adapt-stepindicator="book trip">
  <div adapt-status="complete" aria-labelledby="select flight">
    <a href="uri">Select flight</a>
  </div>
  <div adapt-status="current" aria-labelledby="book hotel">
    <a href="uri">Book Hotel</a>
  </div>
  <div adapt-status="" aria-labelledby="book car">
    <a href="uri">Book car</a>
  </div>
  <div adapt-status="" aria-labelledby="purchase trip">
    <a href="uri">Purchase trip</a>
  </div>
</div>
```

2024年6月27日 Takayuki Shimada

ガイドライン 1.4 判別可能

<https://waic.jp/translations/WCAG21/Understanding/distinguishable.html>

原文:コンテンツを、利用者にとって見やすく、聞きやすいものにすること。これには、前景と背景を区別することを含む。

ざっくりいうと...

原文の通り、見やすく聞きやすいものにすること。代替フォーマットを提供することを述べてるガイドラインもあるけど、このガイドラインはデフォルトで見やすく聞きやすいものになっていることを求めている。

なんで...?

視覚及び聴覚に障害のある利用者は、前景と背景の情報を区別することがはるかに困難である。

達成基準 1.4.1 色の使用を理解する

原文:(レベル A): 色が、情報を伝える、動作を示す、反応を促す、又は視覚的な要素を判別するための唯一の視覚的手段になっていない。

ざっくりいうと...

ある情報を伝えるとき、色だけに依存してはダメだよ。

他の情報(テキスト、十分な明度差、形状、模様、装飾など)も使って提示しようね。

例えば次のようなのはNGだよ

- 「必須項目は赤字」「赤字はエラー」「赤がメアリーの売上、青がトムの売上」

誤解を避けるために: 色を使っちゃダメというわけではないよ。

なんで...?

色は誰にでも知覚できるものではないよ。

- 色覚特性のある人
- ロービジョンは色覚に限界を感じるがよくある
- 年配の利用者の多くも色がよく見えない

色の表現に制約のある場合があるよ。

- モノクロ印刷やモノクロディスプレイ
- 性能の悪い／劣化したディスプレイ

こういう場合でも情報がつたわるようにするためだよ。

どうしたら実現できるの？

「十分な達成方法」なので、他にも実現方法はあるけど一例として。

- 「必須項目は赤字」ではなく、必須項目にはテキストで「必須」と書く
- リンクに下線を引く
- チャートの面に模様をつける

達成基準 1.4.2 音声の制御を理解する

<https://waic.jp/translations/WCAG21/Understanding/audio-control.html>

原文:(レベル A): ウェブページ上にある音声自動的に再生され、3 秒より長く続く場合、その音声を一時停止又は停止するメカニズム、もしくはシステム全体の音量レベルに影響を与えずに音量レベルを調整できるメカニズムが利用できる。

ざっくりいうと...

3秒以上の音声自動的に再生される場合、停止や音量調整ができるようになってなきゃいけないよ。

これは達成基準のなかでも特に重要で、満たされないとページ全体が使いえなくなってしまうことがあるよ。

なんで...？

スクリーンリーダーを使ってる人が困るからだよ。

スクリーンリーダー利用者は音声を聞きながらナビゲートするけど、自動再生する音声にかき消されてしまうかもしれない。

音声が一時的に停止できたとしても、一時停止ボタンまでナビゲートすることすらままならなくなってしまう可能性があるよ。

音声の流れるなか集中することが苦手な人にとっても役に立つよ。

どうしたら実現できるの？

音声は自動再生しないのが一番いい。

利用者が再生ボタンを押したとき、初めて再生される。

自動再生される場合、3秒以内に停止するようにするといいい。

3秒より続く音声を自動再生する場合、一時停止ボタンはページ先頭付近におく。

達成基準 1.4.3 コントラスト (最低限)を理解する

<https://waic.jp/translations/WCAG21/Understanding/contrast-minimum.html>

原文:(レベル AA): テキスト及び文字画像の視覚的提示に、少なくとも 4.5:1 のコントラスト比がある。ただし、次の場合は除く:

大きな文字

サイズの大きなテキスト及びサイズの大きな文字画像に、少なくとも 3:1 のコントラスト比がある。

付随的

テキスト又は文字画像において、次の場合はコントラストの要件はない。アクティブではないユーザインタフェース コンポーネントの一部である、純粋な装飾である、誰も視覚的に確認できない、又は重要な他の視覚的なコンテンツを含む写真の一部分である。

ロゴタイプ

ロゴ又はブランド名の一部である文字には、最低限のコントラストの要件はない。

ざっくりいうと...

テキストと文字画像(画像に含まれる文字)のコントラストをしっかり確保しよう!

サイズの大きなテキスト(24px以上の通常テキスト、18.5px以上の太字テキスト)の場合は3:1、そうでない場合は4.5:1のコントラスト比が必要です

なんで...?

中度のロービジョンの人がテキストを読めるようにするためだよ。

読みやすさに一番寄与するファクターは輝度だから、コントラストの算出には相対輝度が使われるよ。

ちなみに...

APCAも相対輝度を使ってLcを算出してる。

たまにAPCAが「色覚特性を配慮した計算」と紹介されていることがあるけど誤り。

色覚特性があっても、知覚される輝度はあまり変わらないといわれている。

(P型は赤が暗く見えてしまうというのがあるけどここだけ抑えておけばオッケーという認識)

ちなみに...

「サイズの大きなテキスト」の基準は欧文書体における基準。

日本語は同等なサイズの基準を別途定めきやいけないってことになっている。

WAICの現行の翻訳では、拡大教科書の規格に基づき、29.3px以上の通常テキスト、24px以上の太字テキストが日本語における「サイズの大きなテキスト」と訳注している。

...が、それってほんとにそうなの？ というのが最近見直されようとしている。

[JIS X 8341-3:2016のいう「サイズの大きなテキスト」について - 水底の血 \(hatenablog.jp\)](#)

ざっくりいうと、欧文と同じサイズの基準でもよいのではないかと著者momdo氏は考えている。

WAICでも見直しの動きをしていて、すぐじゃないかもしれないけど訳注が変わるかもしれない。

どうしたら実現できるの？

背景色と前景色に十分なコントラストを確保しよう！

2024年8月22日 鈴木亜依 → 嶋岡勝仁

達成基準 1.4.4 テキストのサイズ変更を理解する

<https://waic.jp/translations/WCAG21/Understanding/resize-text.html>

原文: キャプション及び文字画像を除き、テキストは、コンテンツ又は機能を損なうことなく、支援技術なしで **200%** までサイズ変更できる。

意図

> この達成基準の意図は、テキストベースのコントロール ([ASCII などのデータ形式であるテキスト文字に対して] 視覚的に見ることができるように表示された文字) を含む視覚的にレンダリングされたテキストを、例えば画面拡大ソフトのような支援技術を使わずに軽度の視覚障害のある人が、そのまま読むことができるように保証することである。利用者はウェブページ上のすべてのコンテンツを拡大することでメリットを得ることができるが、テキストは最も重要である。

つまり...

テキストは支援技術を使わなくても拡大縮小できるようにしておいてね。
なぜなら、ウェブページ上で拡大できるコンテンツの中で最も重要なのはテキストだから。

ASCIIってなんだっけ？

> [ASCII などのデータ形式であるテキスト文字に対して] 視覚的に見ることができるように表示された文字

→コンピュータに対する2進数で表現するASCIIコードじゃなくて、普通に人間が見るテキストのこと？

→[合ってた](#)

まずはブラウザがやるべきこと

> コンテンツを拡大縮小することは、第一にユーザエージェントが果たすべき役割である。

⌘ ++ or ⌘ +- とかでまずはブラウザが機能を果たしてね。そして我々制作者の責務は、そんな拡大縮小を妨げないこと。(ここではユーザエージェントをブラウザと解釈してます)

> コンテンツ制作者が、ユーザエージェントによってズームサポートを提供していないウェブコンテンツ技術を使用している場合、コンテンツ制作者はこのタイプの機能を提供するか、又はユーザエージェントによって提供される機能のタイプで動作するコンテンツを提供する義務がある。ユーザエージェントがズーム機能を提供していないが、利用者にテキストの大きさの変更するのを許可する場合、コンテンツ制作者は、テキストの大きさを変更するときにコンテンツが利用可能なままであることを保証する責任がある

(ズーム機能が)有るのを使え、無いなら作れ

200%

200%、つまり幅と高さが2倍になるまで拡大可能なら、コンテンツは達成基準を満たしている。
逆にそれ以上でかくなりすぎると、横スクロールバーが発生したり文字が重なったり縦表示になったりしてユーザビリティを損ねる可能性もある。

注記

> 文字画像は、画素に分解されてしまうので、テキストと同じように拡大できないため、可能な限りテキストを用いることを勧める。また、文字画像の前景と背景のコントラスト及び色の組み合わせを変更することを必要とする利用者もいるが、テキストよりも困難である。

キーワードで絞り込む

こういうのやめよう。

メリット

ロービジョンの人に役立つ！

事例

- 視覚障害のある利用者は、ブラウザにあるウェブページのテキストの大きさを **1 em から 1.2 em** に増やしている。その利用者は、小さいテキストを読むことはできないが、大きいテキストは読むことができる。テキストに大きなフォントを使用すると、ページ上のすべての情報が表示される。

達成するためには？

CSSでスタイルを当てるとき、テキストを em とか % とか名前 (largeとかx-largeとか) とか、相対的な大きさでしているべし！

失敗例

こんな感じで失敗することもあるから気をつけてね

<https://waic.jp/translations/WCAG21/Techniques/failures/F69#example-1>

まとめ

とにかくめちゃくちゃ文量が多い……

でもそれだけ、テキストは大事ということ。

アスキーアートはきつい

```
( 'ω)
γ/ γ^ (';ω;`) ウツ...
/| 、イ( ^ ^
.| | | } ) \ _ _ \ \>>1
{ | | | ^ ^ ^ ^ ^ ^ ^ ^
.\ | | T "" --- ^ ^ ^ ^ ^
| | | / / / / /
```

内藤ホライゾンとか。

スクリーンリーダー使ってる人って、文字打ち込むとき、顔文字とか記号(♪とか♥とか)ってどうしてるんだろう？

```
      /  ^  \
    cニニニ( ^ω^ )ニッ
      |    /      ブーン
    (  \ノ
      />ノ
    三   レレ
```

こんな感じでrole="img"として表現するパターン

<https://www.w3.org/TR/html-aria/#example-13>

こういうのとかもimgとして見せちゃうのがえんちゃう

<https://twitter.com/uhyo/status/1825351678012612894>

達成基準 1.4.5 文字画像を理解する

原文: 使用している技術で意図した視覚的提示が可能である場合、文字画像ではなくテキストが情報伝達に用いられている。ただし、次に挙げる場合を除く:

カスタマイズ可能

文字画像は、利用者の要求に応じた視覚的なカスタマイズができる。

必要不可欠

テキストの特定の表現が、伝えようとする情報にとって必要不可欠である。

※ロゴタイプ (ロゴ又はブランド名の一部である文字) は必要不可欠なものであると考えられる。

意図

特定の視覚的なスタイルで文字を見たい人が、そのスタイルを自由に調整できるようにすること。つまり、フォントのサイズや色、背景色やフォントの種類、行の間隔、文字の配置などを、ユーザーが自身の好みに合わせて変えられるようにすることを我々に勧めている。これにより、見やすさや読みやすさを自分で調整しやすくすることを目指している。

テキストでも問題ないなら、テキストを使おう

コンテンツ制作者が同じ視覚的な効果を達成するためにテキストを使用できる場合、コンテンツ制作者は情報を提示するのに画像を用いるのではなく、テキストを用いることが望ましい。

> 利用者が文字画像を自分の好みに合わせてカスタマイズできる場合にも、文字画像を用いてもよい。

とあるけど、難しそうな気がする。

文字が含まれていても、グラフ、スクリーンショットやダイアグラムなどは、文字以外のものを通じて重要な情報を視覚的に伝えているから、文字画像には含まれない。

[文字画像 \(例外なし\)](#) みたいなものもあるので混乱する――

メリット

- ロービジョンの利用者 (コンテンツ制作者の指定した書体、サイズ、及び／又は色では、テキストが読みづらいことがある)
- 視線移動に問題のある利用者 (コンテンツ制作者の指定した行間及び／又は配置では、テキストが読みづらいことがある)
- 読字に影響を及ぼす認知の障害のある利用者

これがなんでメリットになるんだろう？

視線移動に関しては、以前輪読会で出た「川」が影響するから？だから文字間とかを意図的に調整することで「川」のように見えづらくするとか？

それ以外は、文字画像のメリットの説明としてはよく分からなかった。

↑文字画像を使わないことによるメリットを言ってる

達成方法

1. [C22: テキストの視覚的提示を制御するために、CSS を使用する](#)
2. [C30: テキストを文字画像に置き換えるために CSS を利用し、切り替えのためのユーザインタフェースコントロールを提供する](#)
3. [G140: 異なる提示を可能にするために、情報と構造を表現から分離する](#)
4. [PDF7: スキャンされた PDF 文書で OCR を実行し、実際のテキストを提供する](#)

ユーザー側で切り替えられるようにしよう！

力尽きたっちゃ！

2024年10月17日 嶋岡勝仁

達成基準 1.4.6 コントラスト (高度) を理解する

達成基準
(レベル AAA)

テキスト及び文字画像の視覚的提示に、少なくとも 7:1 のコントラスト比がある。ただし、次の場合は除く:

大きな文字

サイズの大きなテキスト及びサイズの大きな文字画像に、少なくとも 4.5:1 のコントラスト比がある。

付随的

テキスト又は文字画像において、次の場合はコントラストの要件はない。アクティブではないユーザインタフェース コンポーネントの一部である、純粋な装飾である、誰も視覚的に確認できない、又は重要な他の視覚的なコンテンツを含む写真の一部分である。

ロゴタイプ

ロゴ又はブランド名の一部である文字には、最低限のコントラストの要件はない。

大きなサイズって？

<https://waic.jp/translations/WCAG21/#dfn-large-scale>

少なくとも 18 ポイント、又は 14 ポイントの太字。あるいは、中国語、日本語、及び韓国語 (CJK) のフォントは、それと同等の文字サイズ。

→ 多くの主流となっている本文テキストで用いられるフォントにおいて、14 ポイントと 18 ポイントは、1.2em と 1.5em、又は、(本文フォントが 100% であると仮定して) デフォルトサイズの 120% と 150% に、おおよそ同等である。

→ この達成基準について評価する時には、文字サイズのポイント数は、ユーザエージェントから取得されるべきであり、又はユーザエージェントが行うフォントの計算基準に基づいて計算するべきである。ポイント数は、CSS3 Values で定義されている CSS の pt サイズに基づいている。ポイント数と CSS ピクセルの比は $1\text{pt} = 1.333\text{px}$ であり、したがって 14pt と 18pt は約 18.5px と 24px に相当する。

意図

中度のロービジョンの人がテキストを読めるように、テキストとその背景との間に十分なコントラストを提供すること

読解力で評価すると、色覚異常ではない人にとって、色相及び彩度は文字の読みやすさにほとんど又は全く影響がない

色覚異常は、輝度コントラストに多少の影響を及ぼす可能性がある。

色覚異常の人でもテキストと背景との間の適切なコントラストを持つように、色が主要因とならないような方法でコントラストを算出している。

色以外にコントラストってあるの？

→ コントラストは主に色情報に基づいて考えられることが多いですが、色以外にも次のような要素がコントラストの一部として考慮されることがあります：

1. *明度コントラスト*：色の明るさ(輝度)に基づくコントラストです。これは白黒のグレースケールで表現された場合のコントラストに相当します。

2. *テクスチャコントラスト*：異なるパターンや質感の組み合わせによるコントラストです。視覚的に異なるテクスチャの組み合わせによってもコントラストを感じることができます。

3. *形状コントラスト*：異なる形やサイズの要素間に生じるコントラストです。形状の違いによって対象物を認識しやすくします。

4. *動きのコントラスト*：動いている要素と静止している要素との間で生じるコントラストです。

色覚異常の方に配慮したデザインでは、明度やテクスチャ、形状を利用してコントラストを強調し、色の違いだけに頼らない方法が推奨されます。

より大きい及びより広い文字ストロークを持つテキストは、より低いコントラストでも読みやすい。したがって、より大きいテキストに対するコントラスト要件は、より低い。

前述のテキストに対するコントラストの要件は、達成基準 1.4.5 で述べられているように、文字画像(ピクセルでレンダリングされ、画像フォーマットで保存された文字)にも適用される。

コントラスト比を含めた論理的根拠と計算式に関する注記はむずすぎたので割愛・・・

メリット

- ロービジョンの人は、背景とのコントラストがないテキストを読むのが困難なことがよくある。これは、コントラストをさらに低下させる色覚異常がある人の場合、深刻となりえる。テキストとその背景との間に最低限のコントラスト比を提供することで、たとえその人があらゆる色を見ることができなくても、テキストをより読みやすくすることができる。また、まれに全く色が見えない人にも有用である。

失敗例

<https://waic.jp/translations/WCAG21/Techniques/failures/F24>

全部の背景色、前景色を指定しないとダメっぽい？

keelaiもそう言ってた

<https://lifull.slack.com/archives/C02JQB09W86/p1729081548752249>

達成基準 1.4.7: 小さな背景音、又は背景音なしを理解する

<https://waic.jp/translations/WCAG21/Understanding/low-or-no-background-audio.html>

達成基準

(レベル AAA): 収録済の音声しか含まないコンテンツで、(1) 前景に主として発話を含み、(2) 音声 CAPTCHA 又は音声ロゴではなく、かつ、(3) 例えば、歌やラップなどのように、主として音楽表現を意図した発声ではないものについては、次に挙げる事項のうち、少なくとも一つを満たしている:

背景音なし

音声は背景音を含まない。

消音

背景音を消すことができる。

20 デシベル

背景音は、前景にある発話のコンテンツより少なくとも 20 デシベルは低い。ただし、継続時間が 2 秒以内で発生頻度が低い背景音は除く。

音声ロゴとは

タラッタッタター♪ で、マクドナルドを想起するようなあれ。

ナーナナナナッナナー♪ で、アルトリを想起するようなあれ。

意図

この達成基準の意図は、発話ではないあらゆる音が、音声の聞こえづらい利用者が発話を背景音又は前景にある不要な他の発話コンテンツと区別することができるようにすることである。

メリット

- 音声の聞こえづらい利用者は、発話と背景音を区別しづらいことがしばしばある。

2024年11月14日 千葉穂乃美

<https://jira.next-group.jp/wiki/x/QdZsGg>

2024年10月17日 Takuya Nakajima

達成基準 1.4.11: 非テキストのコントラストを理解する AA

ざっくりいうと？

非テキストの要素、特に重要なグラフィックとUIは、3:1以上のコントラスト比を保とうねという話
グラフィックにはアイコンとか、グラフとかが含まれる

なんで...？

中程度のロービジョンの人など、視覚に困難を抱える人々がウェブ上の情報や操作をより確実に理解・操作できるようにするため

例外は...？

- 情報を伝えるうえで特有のグラフィックの提示が必要不可欠な場合(例: ブランドロゴ、旗など)には、コントラストの要件が適用されません。
- 非アクティブなUIは対象外になる(disabledとかaria-disabledでそう表現されてればaxeとかも素通りする)

たとえばどこを気をつければいい？

- チェックボックスの中のチェックとか
- 開閉情報を示す矢印と背景色とか

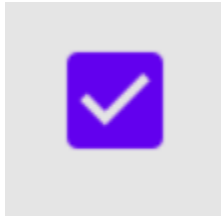
細かいはなし

この基準は主にコントロール自身の色をさしていてコントロールの境界線について言及してないんだけど、コントロールの境界線がそのコントロールを識別する唯一の方法の場合は境界線に十分な境界線をもとめる(したみたいなケース)

Button

Button

チェックボックスのチェックとかスライダーのつまみみたいに状態をとくていするために必須の要素はコンポーネントの内部にあるかもしれない、その場合確認すべきはその要素とコンポーネントの塗りの色となる



達成基準1.4.1(色のみが情報や動作などを識別する唯一の視覚的な手段になってはいけない)とあわせて判断しなければいけないケースがよくある



上記の図は上の例は色だけが唯一の判別手段になっている。後者は色もそうだけど背景とのコントラストもたりてない

達成基準 1.4.12: テキストの間隔を理解する

ざっくりいうと？

この基準は、行の間隔、段落の後のスペース、文字間、単語間のスペースを適切に調整することで、読みやすいコンテンツを維持することを目的としています。

次のスタイル設定をしてもコンテンツが読めなくなったり、UIがうごかなくなったりしないよねって話

line-heightを1.5以上、p+*のmargin-topをfont-sizeの2倍、letter-spacingをfont-sizeの0.12倍、(単語の間隔をfont-sizeの0.16倍に設定する: 英語圏特有かな)

コードでいうとこんな感じ

```
* {  
  line-height: 1.5em !important;  
  letter-spacing: 0.12em !important;  
}  
p + * {  
  margin-top: 2em !important;  
}
```

これはこのスタイルをサイト側が適用してくださいという話ではなく、利用者が拡張とか使って適用しても崩れないことを求めています

なんで...?

この基準は、ロービジョンや失読症の人々を含む、閲覧者がより快適にテキストを読むことができるようにすることを目指しています。テキストの間隔を個々のニーズに合わせて調整可能(chrome extensionとかで)することで、読解速度の向上や視覚上のストレス軽減を図ります。

例外は...?

- 使用してるスタイル技術が上書き可能である時の話をしてるので次の場合は対象外となる
 - PDFとか
 - 動画内のキャプション
 - 文字画像 (canvasも含め)

NG事例

<https://codepen.io/Takuya-Nakajima/pen/pvzoMmJ>

達成基準 1.4.13: ホバー又はフォーカスで表示されるコンテンツを理解する

ざっくりいうと？

キーボードフォーカスとかhoverで表示されたりするコンテンツがある場合はフォーカスを外したりカーソル移動したりせずに閉じられる方法を求める(例えばescキーでとじるとか)
またホバーでコンテンツがでる場合、コンテンツにカーソルがのってもコンテンツが閉じないこと
表示されたコンテンツが何もしていないのに勝手に消えたりしないこと

なんで...？

目的は、追加コンテンツがユーザーの意図せずに発生してタスクを邪魔しないようにし、またはユーザーが追加コンテンツに気づかないことがないようにすることです。

特に視覚障害者やマウス操作が困難なユーザーにとって、フォーカスやホバーによる予期しない変化は大きな困難をもたらすため、これを回避するための要件を設けています。

例外は...？

- 例外として、ユーザーエージェントによって完全に制御され、コンテンツ制作者が変更できない場合、例えばHTMLの`title`属性を使ったブラウザのツールチップなどは、この基準から除外されます。
- 追加されるコンテンツがフォーカスをうけとるような場合(例えばdialogとか)はそもそもhoverとかフォーカスで起動させるべきではないのでスコープ外

対応事例

<https://waic.jp/translations/WCAG21/Understanding/content-on-hover-or-focus.html#examples>

1. 非表示にすることができるツールチップ - ユーザーが`Esc`キーを押すことで簡単にツールチップを消去できるデザイン。
 2. ホバーすることができるツールチップ - 大きなポインタを使用してもツールチップを隠さず、コンテンツを確認することができるデザイン。
-

2025年1月9日 鈴木亜依

達成基準 2.1.1: キーボードを理解する

達成基準

レベル A

概要

コンテンツのすべての機能は、特定のタイミングやキーストローク(※1)を必要とせず、キーボードインタフェースを通じて操作をできるようにする。

- ただし、キーボードから合理的に操作できないものは除く(※2)
- 入力の代替として利用者の動作に依存する操作が「できる」ことは問題ない。根本の機能が利用者の動作に依存した操作を必要としていなければいい(※3)
- テンキーを使って操作するマウスキーの使用は、キーボードに相当しないので、この達成基準を満たさない

※1: キーストロークを必要としない

キーストローク=キーボードのキーを押すこと。

利用者が短時間のうちに複数のキーストロークを繰り返す又は長い間キーを押し続けなければならない状況が挙げられる

※2: キーボードから合理的に操作できないものは除く

例としては以下のようなものが挙げられる

- 軌跡に依存した入力

自由な手描き、水彩画、及び障害物コースを通るヘリコプターの飛行

- 軌跡に依存しない入力

直線や規則的な幾何学的図形を描く、ウィンドウのサイズを変更する。ある位置へオブジェクトをドラッグする

※3: 根本の機能が利用者の動作に依存した操作を必要としていなければいい

これは根本的な機能に関するものであり、入力手法に関するものではない

例: テキスト入力に手書き入力を用いるのであれば、その入力手法 (手書き) は利用者の動作による軌跡に依存した入力が必要とするが、

その根本的な機能 (テキスト入力) は利用者の動作による軌跡に依存した入力が必要とするものではない。

意図

可能な限りコンテンツは「キーボードインタフェース」で操作ができるようにすること

- ・「代替キーボード」、「キーボードエミュレータの機能を果たす入力デバイスを使用しなければならない人」でも操作できるようになる。
- ・ポインタを目で追うのが困難なロービジョンの人は、キーボードで操作できるとソフトウェアの使用がはるかに容易になる（又は、単に使えるようになる）。

メリット

以下のような人が利用しやすくなる

全盲の人

目と手の協調運動を必要とするマウスのようなデバイスを使用できない人

ロービジョンの人

画面上のポインタを見つけたり、目で追ったりするのが困難である人

手に震えのある人

マウスを使うのが非常に難しいと感じ、通常はキーボードを使用している人

達成基準 2.1.2: キーボードトラップなしを理解する

達成基準

レベル A

概要

コンポーネントにあたったキーボードフォーカスをキーボードインタフェースだけを用いて外すことができる。

- ウェブサイトの閲覧時、キーボードを使ってナビゲーションしている際に、フォーカスが特定の領域に「閉じ込められてしまう」状態のこと。
例: モーダルウィンドウを開いた際に【ESC】キーで元のコンテキストにフォーカスを移動できなくなる。
- 標準的な方法(※)でフォーカスを外せない場合は、フォーカスを外す方法を利用者に通知する
※修飾キーを伴わない矢印キー、Tab キー、又はフォーカスを外すその他の標準的な方法

意図

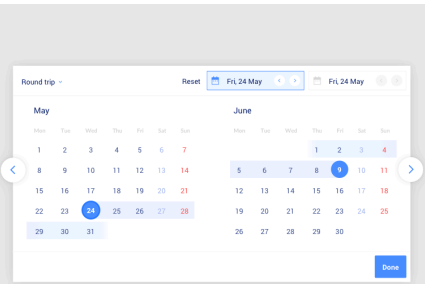
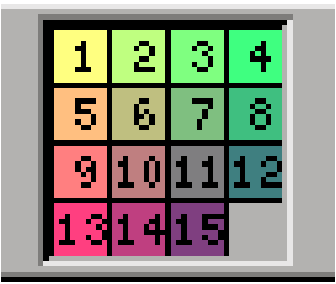

キーボードフォーカスを「トラップ」しないことを確実にするためのものである

- ・複数のフォーマットが 1 ページ中に組み合わせられ、かつ、プラグイン又は埋め込みアプリケーションを使用してレンダリングされるときによく起こる問題である。
- ・ポインタを目で追うのが困難なロービジョンの人は、キーボードで操作できるとソフトウェアの使用がはるかに容易になる (又は、単に使えるようになる)。

メリット

キーボード又はキーボードインタフェースに依存している人の利用が容易になる
全盲の人及び物理的な障害のある人など

事例

		
<p>カレンダーウィジェット</p>	<p>パズルアプレット</p>	<p>モーダルダイアログボックス</p>

ウィジェット内のコントロールは、ウェブページ内のタブ移動順序の一部であり、利用者はウィジェット内のコントロールだけでなく、その後続く任意のリンク又はコントロールにも Tab キーで移動できる。	Tab キー及びその他のキーストロークはアプレットによって処理される。アプレットから抜け出すのに用いるキーストロークを説明する指示は、アプレット自身の中だけでなく、アプレットの前にも提供される。	フォーカスはダイアログ内に閉じ込められる。ダイアログの最後のコントロールから Tab キーで、最初のコントロールにフォーカスが移動する。「キャンセル」又は「OK」ボタンをアクティブにすると閉じられる。
--	---	--

2025年1月23日 千葉穂乃美

<https://jira.next-group.jp/wiki/x/SyDxGg>

2025年2月6日 Takuya Nakajima

- [達成基準 2.2.2: 一時停止、停止、非表示を理解する](#)
- [達成基準 2.2.3: タイミング非依存を理解する](#)
- [達成基準 2.2.4: 割り込みを理解する](#)

達成基準 2.2.2: 一時停止、停止、非表示を理解する

<https://waic.jp/translations/WCAG21/Understanding/pause-stop-hide.html>

原文

動きのある、点滅している、スクロールする、又は自動更新する情報は、次のすべての事項を満たしている

動き、点滅、スクロール

動きのある、点滅している、又はスクロールしている情報が、(1) 自動的に開始し、(2) 5 秒よりも長く継続し、かつ、(3) その他のコンテンツと並行して提示される場合、利用者がそれらを一時停止、停止、又は非表示にすることのできるメカニズムがある。ただし、その動き、点滅、又はスクロールが必要不可欠な動作の一部である場合は除く。

自動更新

自動更新する情報が、(1) 自動的に開始し、(2) その他のコンテンツと並行して提示される場合、利用者がそれを一時停止、停止、もしくは非表示にする、又はその更新頻度を調整することのできるメカニズムがある。ただし、その自動更新が必要不可欠な動作の一部である場合は除く。

ざっくりいうと...

ウェブコンテンツ上の動きや点滅、スクロールする情報に関するガイドラインです。ユーザーがこれらを一時停止、停止、非表示にできる手段を提供しなければならないというものです。

対象

動き、点滅、スクロール

目に見えるコンテンツが動きの感覚を伝えているコンテンツのことを指している。

一般的な例は、動画、同期したメディアによる提示、アニメーション、リアルタイムのゲーム、スクロールする株価表示などがある。

自動更新

あらかじめ設定された間隔で更新したり、消えたりするコンテンツのことを指している。

自動で動くカルーセルとか

なんで...?

動きや点滅する要素があることで、多くの人がウェブサイトを使うことが難しくなることがあります。具体的には、注意散漫になりやすい、視覚的な情報に敏感な、または特定の障害を持つ人々にとって不便を感じる可能性があります。

これを防ぐための配慮として一時停止や停止、非表示にできるオプションを提供することが求められます。

再開後の動きについて

一時停止したコンテンツは再開時に、「利用者が一時停止したところから再生を続ける」か「最新の状態で再生を始める」のいずれかになる

リアルタイムコンテンツでない場合、コンテンツを読むために一時停止したユーザーにとっては停止箇所からスタートするのが最適である

一方、気象レーダーや株価表示、交通情報カメラとかは再開時に停止中の位置からスタートすると古い情報を最新のものと勘違いしてしまう恐れがあるので最新の表示に移行することが好ましい

例外は...?

重要な例外として、動きが「5秒以内」であれば一時停止や停止の機能を提供する必要はありません。

たとえば、短いアニメーションや移り変わる内容が自然に停止する場合はそれに該当します。

対応事例

- 自動スクロールするテキストがある場合、ユーザーが「停止」ボタンや「一時停止」ボタンをクリックできるような設定を実装する
- 点滅する広告がある場合、広告自体に表示を止める機能や、ユーザーが自由に消すことができるボタンを設ける
- スライドショーがある場合、再生・停止を制御するUIを提供する
- 株式相場表示には「一時停止」と「再開」のボタンがあり、再開時に現表示が最新でないことを伝える。また最新の株価表示まで進めるボタンを用意する

達成基準 2.2.3: タイミング非依存を理解する AAA

<https://waic.jp/translations/WCAG21/Understanding/no-timing.html>

原文

タイミングは、コンテンツによって提示されるイベント又は動作の必要不可欠な部分ではない。ただし、インタラクティブではない同期したメディア及びリアルタイムのイベントは除く。

ざっくりいうと...

ウェブコンテンツにおいて、ユーザーの操作や情報取得にタイミングの制約を設けないようにするためのガイドライン。

ユーザーが自分のペースで利用できるように設計する必要があります。

なんで...?

時間制限があると、読むことに時間がかかったり、操作に時間がかかったりするユーザー、繊細な操作が必要なデバイスを利用しているユーザー、あるいはスピードに依存しがたい状況にいる人々が困ってしまう。

タイミング制約を緩和することで、誰もが公平にコンテンツを利用できるようにするのが目的

例外は...?

リアルタイムイベント(オークションとかライブパフォーマンスのウェブ放送とか、VR空間上での人々のやりとりとか)

対応事例

- 試験において回答時間が採点結果に影響を与えなくする

- リアルタイムなバトルではなく、ターン制を採用したゲーム

達成基準 2.2.4: 割り込みを理解する AAA

<https://waic.jp/translations/WCAG21/Understanding/interruptions.html>

原文

割り込みは、利用者が延期、又は抑制することができる。ただし、[緊急](#)を要する割り込みは除く。

ざっくりいうと...

ウェブコンテンツにおけるユーザーの体験を中断しないように設計するためのガイドラインです。ユーザーの操作やタスクを中断する可能性のある要素について、必要でない場合は回避するか、ユーザーが制御できる手段を提供することが求められます。

なんで...?

中断があることで、ユーザーが作業に集中できなくなったり、タスクを完了するのが困難になる可能性がある。

特に集中持続が難しい人や、認知の問題を持つ人々にとっては大きな障害となり得る。故に中断を回避するための対策を講じる必要がある。

何が対象になっているか

- ポップアップメッセージや通知
- システムアラートや警告メッセージ

例外は...?

セキュリティや安全にかかわる重要な情報や、緊急の警告などの中断は許容される。これらはユーザーに即時の注意を喚起する必要があるため、中断は不可避である。

やっちゃってる事例

- n秒後にリダイレクト(あるいはリフレッシュ)します みたいなやつを利用している

対応事例

- システムメッセージや通知がある場合、ユーザーが自分のタイミングで確認できるようにバッジ表示やアイコンの変更を使う。
- 突然のポップアップ広告などは避け、ユーザーが自発的に起動できるようにする。

- チャットウィンドウや通知メッセージは、ユーザーが表示タイミングを制御できる設定を用意する。

2025年3月6日 嶋岡勝仁

達成基準 2.2.5: 再認証を理解する(レベル: AAA)

<https://waic.jp/translations/WCAG21/Understanding/re-authenticating.html>

原文

認証済のセッションが切れた場合は、再認証後でもデータを失うことなく利用者が操作を継続できる。

ざっくりまとめ

この達成基準の意図は、処理を完了する前に利用者をログアウトさせてしまった場合等に、全ての利用者が完了できるようにすること。

セキュリティ上の理由で、多くのサイトが何の操作もせずに一定時間が経過した後の認証の制限時間を設定している。

処理中にログアウトさせられても、再度ログインした際に、すでに入力済みのデータを失うことなくその処理を継続できるようにすることが重要である。

その理由とメリット

この達成基準は、作業を完了するのに時間がかかる利用者の役に立つ。

- 認知能力の低下している利用者は、ゆっくり読むので、アンケートを読んで回答するのに時間がかかる場合がある。
- スクリーンリーダーを使用している利用者は、複雑なフォームを操作して完了させるのに時間がかかるかもしれない。
- 運動障害のある利用者または代替入力デバイスで操作する利用者は、フォーム内を操作して入力を完了させるのに、やはり時間がかかることがある。
- 手話通訳者がろう者に音声コンテンツを通訳しているような状況では、制限時間を調整することも重要である

事例(原文)

- ショッピングサイトの支払手続
手をほとんど使うことのできない利用者が、ショッピングサイトにログインしている。アプリケーションにクレジットカード情報を入力するのに時間がとてもかかるので、利用者が支払手続をしている間に制限時間が切れる。利用者が支払手続に戻ってフォームを送信するとき、サイトは再認証するためのログイン画面を表示する。利用者がログインした後、支払手続は同じ段階の同じ情報に復元される。セッションが時間切れになってもサーバが一時的に送信を受け付けて保存してくれていたため、利用者はデータを失わずに済み、再認証が完了した後も同じ状態に戻されたのである。
- 電子メールプログラムでの認証
電子メールプログラムは、30 分後に認証が時間切れになる。そのプログラムは、時間切れの数分前に利用者にプロンプトを表示し、再認証するために新しいウィンドウを開くリンクを提供する。作成中のメールがあった元のウィンドウはそのまま、再認証後に、利用者はそのデータを送信することができる。
- 制限時間のあるアンケート
1 ページのウェブページで提供されている長いアンケートは、ページの冒頭に、15 分経過後にセッションが時間切れになることを知らせる情報がある。アンケートはどの時点でも保存することが可能で、後から完了させることも可能であることも、利用者には知らされている。そのウェブページには、部分的に記入が終わったフォームを保存するためのいくつかのボタンがある。さらに、アクセシビリティ サポートされたウェブコンテンツ技術の一覧にある信頼されている JavaScript によって、セッションが時間切れに近づいたらポップアップで通知されるように利用者が選択できる。

ダメな例

一定時間後にログアウトされた(セッションが切れた)あと、入力したデータが残っていないパターン。

- データ入力後に一定時間経過し、ログアウトされている状態のまま、その画面でフォーム送信すると、再度ログインが促され、ログインページなどに遷移させられる。データは処理されておらず、ログイン後もまた最初からやり直さなければならない。
- データ入力後に一定時間経過し、ログアウトされている状態のまま、その画面でフォーム送信すると、再度ログインが促され、ログインページなどに遷移させられる。ログイン後入力画面に戻っても、フォームには入力したデータは残っておらず、利用者は再入力しなければならない。
- 入力画面で誤って別ページに遷移した後(ブラウザバック含む)、再度入力画面に戻ったときに、入力したデータが残っていない、などもあるかも。

感想

これシングルAでもよくね？

達成基準 2.2.6: タイムアウトを理解する(レベル: AAA)

<https://waic.jp/translations/WCAG21/Understanding/timeouts.html>

原文

データの損失を引き起こす恐れのある**利用者の無操作**の残り時間が警告される。ただし、利用者が 20 時間以上何もしなくてもデータが保持される場合は、この限りではない。

ざっくりまとめ

原文

この達成基準の意図は、タイムアウトが用いられるときに、どの程度の無操作時間によってページがタイムアウトしてデータが失われるのかを、利用者が確実に把握することである。認知障害のある利用者にとっては、コンテンツを閲覧したり、オンラインフォームを全て記入するなど機能を実行したりするのに時間がかかることがあるため、時間制限のあるイベントの使用は大きな障害となる可能性がある。

ホテルの部屋の予約や航空券の購入などのオンラインプロセスを完了するまでの間に、認知障害のある利用者は、プロセスを完了するために必要な長い指示及びデータ入力に圧倒されることがある。利用者は、一度にプロセスを完了することができず、休憩を取る必要がある場合がある。利用者は、プロセス内の現在位置を見失うことなく、かつ、すでに入力された情報を失うことなく、プロセスをそのままにできるようにすべきである。利用者が休憩を取ってから作業内容を確認できない場合、多くの利用者がタスクを正しく完了できなくなる。

この達成基準は、達成基準 2.2.1 タイミング調整可能 と連携して機能するが、特に利用者の無操作状態に関連するタイムアウトの通知に重点を置いている。

この達成基準に適合するための最善の方法は、利用者が入力したデータを少なくとも 20 時間保持することである。これにより、障害のある利用者や高齢者は、必要に応じて休憩を取りながら、タスクを開始し終了することができる。しかし、利用者が入力したデータを保存することが現実的でない場合、コンテンツ制作者は、タイムアウトに至る無操作状態の残り時間について利用者に警告しなければならない。タイムアウトは、関連するタスク又はプロセスの開始時に 1 回だけ表示し、各ステップでは表示する必要はない。

この達成基準は、コンテンツ提供者の関知又は制御の範囲内にあるタイムアウトにのみ適用される。例えば、利用者がウェブブラウザ又はデバイスを閉じて、それまで開いていたページの未送信のコンテンツを失った場合、達成基準に反していることにはならない。

まとめ

- どのくらいの時間経過によってタイムするのかを明示してあげよう。
- 認知障害などで入力に時間のかかるユーザーは、休憩を取る必要がある場合がある。
- 利用者がプロセス内の現在位置を見失うことなく、かつすでに入力された情報を失うことなく、プロセスをそのままにできるようにすべき。
- この達成基準は、どれぐらい無操作状態が続いたらタイムアウトするのかの通知に重点を置いている。
- この達成基準に適合するための最善策は、データを少なくとも20時間保持することである。
- コンテンツ制作者は、タイムアウトに至る無操作状態の残り時間について警告しなければならない。
- この達成基準は、コンテンツ制作者の感知しない、利用者がブラウザやデバイスを閉じて、それまで開いていたページの未送信の入力内容を失ったものについては適用されない。(利用者の意思が含まれているから)

その理由とメリット

この達成基準は、無操作に起因するタイムアウトが利用者に確実に通知されるようにすることで、利用者を支援する。

利用者がタスク実行に許されている残り時間を知っている場合、利用者は必要な休憩を取りつつ作業をやり直すことなく再開できるかどうかを知ることができる。

タイムアウトによる作業の損失に対するフラストレーションが軽減される。

この達成基準は、次のような様々な認知障害を持つ人に役立つ。

- 言語に関連する障害
- 記憶に関連する障害
- 注意及び集中に関連する障害
- 物事の実行機能及び意思決定に影響を与える障害

事例(原文)

- 電子商取引サイトで商品を購入する際、利用者が入力した情報は 20 時間以上保存される。これにより、しばらく作業を中断する必要がある場合でも、戻ってきたときに購入を継続できる可能性が高くなる。
- 納税申告書の提出を可能にするウェブアプリケーションは、セキュリティ上の理由でアプリケーションがタイムアウトすることを通知する。この通知は、1 時間を超える連続した時間にわたって操作がない場合、タイムアウトプロセスが開始されることを示している。
- オンラインの問い合わせフォームには、タイムアウト処理は実装されていない。問い合わせフォームに入力された情報はいつでも送信でき、情報が失われるのは利用者がブラウザウィンドウを閉じた場合のみである。

感想

20時間の根拠は？

keelailに聞いても分からなかった。

- <https://liful.slack.com/archives/C02JQB09W86/p1741233559479499>
- <https://liful.slack.com/archives/C02JQB09W86/p1741234070219889>

20時間の根拠

<https://waic.jp/translations/WCAG21/Understanding/timing-adjustable.html#:~:text=%E4%B8%80%E6%97%A5%E3%81%AB%E8%B5%B7%E3%81%8D%E3%81%A6%E3%81%84%E3%82%8B%E6%99%82%E9%96%93%E3%82%88%E3%82%8A%E3%82%82%E9%95%B7%E3%81%84%E3%81%AE%E3%81%A7%E3%80%81%E4%B8%8A%E9%99%90%E3%81%A8%E3%81%97%E3%81%A6%2020%20%E6%99%82%E9%96%93%E3%82%92%E9%81%B8%E3%82%93%E3%81%A0%E3%80%82>

ガイドライン 2.3: 発作と身体的反応を理解する

<https://waic.jp/translations/WCAG21/Understanding/seizures-and-physical-reactions.html>

原文

発作や身体的反応を引き起こすようなコンテンツを設計しないこと。

意図(原文)

発作性障害のある利用者は、閃光を放つ視覚的なコンテンツによって発作を引き起こすことがある。ほとんどの利用者は、発作に襲われるまでは自分がこの障害を持っていることに気づかない。1997年に、日本でテレビのアニメ番組が700人以上の子どもたちを病院に搬送させる事態を招き、そのうち約500人が発作を引き起こした。特に、警告を実際に読むことができない子どもの利用者に警告はよく見逃されるので、その警告はあまり効果がない。

このガイドラインの目的は、WCAG 2.0に適合しているとされるコンテンツが、1～2秒間であっても発作を引き起こす可能性が最も高い閃光の種類を避けることである。

ポケモンショック

<https://ja.wikipedia.org/wiki/%E3%83%9D%E3%82%B1%E3%83%A2%E3%83%B3%E3%82%B7%E3%83%A7%E3%83%83%E3%82%AF>

当時我々はドンピシャの世代であった(見ていたかは覚えていない…)

2025年4月3日 秦那実

<https://jira.next-group.jp/wiki/pages/viewpage.action?pageId=461244891>

2025年4月17日 千葉穂乃美

https://jira.next-group.jp/wiki/x/e2l_Gw

2025年6月12日 鈴木亜依

達成基準 2.4.3: フォーカス順序を理解する

達成基準

レベル A

概要

ウェブページを順を追ってナビゲートできて、
そのナビゲーション順が意味又は操作に影響を及ぼす場合、
フォーカス可能なコンポーネントは、意味及び操作性を損なわない順序でフォーカスを受け取る
キーボードインタフェースを用いて (一つの要素から次へ) フォーカスを移動するために、定義された順序でナビゲートすること。

意図

利用者がコンテンツをナビゲートする際に、意味と操作性が一致するように情報が提供されることを保証することです。

1. フォーカス可能な構成要素は、操作のしやすさや内容の理解に影響を与える場合に、意味と操作性を保つ順序でフォーカスを受ける必要がある。
2. 意味と操作性を保つ複数の順序がある場合がある。
3. 意味と操作性を保つ複数の順序がある場合、それらのうち一つは提供される必要がある。

詳細

- 利用者のコンテンツに対するメンタルモデルを一貫したものとし、利用者が困惑する可能性を引き下げる。
- 複数の論理的順序が存在する場合もある。例：テーブルの行で移動する、列で移動するなど。これはどちらでも達成基準を満たすことができる

- ナビゲーションの順序は、コンテンツで用いるウェブコンテンツ技術によって定義されている。

動的HTMLでは、tabindex属性を利用して順序を調整できます。加えて、矢印キーを用いたツリーナビゲーション(※1)も含まれます。

- フォーカス順序が音声読み上げ順序と一致しない場合、利用者がウェブページを理解・操作しにくくなる可能性があります。

晴眼のキーボードの利用者とスクリーンリーダーの利用者、フォーカス順序が双方の利用者にとって意味が通じるようにし、どちらかの利用者がランダムに飛び回るようなことのないように注意すべきである

- コンテンツ作成者は、すべてのユーザーを考慮してウェブページを設計することが重要です。

※1

上下の矢印キーを使って、ツリーのノードからノードへと移動することができる。右向き矢印キーを押下するとノードを展開して、それから下向き矢印キーを使用すると新しく展開されたノードへと移動していく。アイテムが展開されたり閉じられたりするたびに、ナビゲーション順序に追加されたり削除されたりするので、このナビゲーション順序は、ツリーコントロールで予期される順序に従っている。そして、アイテムが展開されたり閉じられたりするたびに、ナビゲーション順序に追加されたり削除されたりする。

メリット

文章を順を追ってナビゲートし、フォーカスの順番が読み上げの順番と同じであることを期待しているキーボード利用者に役立ちます

- 論理的で使いやすいフォーカス順序は、キーボードでページを操作する運動障害のある人に役立ちます。
- - 字を読むのが難しい人は、Tabキーで予想外の場所にフォーカスが移動すると混乱しますが、論理的なフォーカス順序が助けになります。
- - 視覚障害のある人も、フォーカスが予期せぬ場所に移動すると迷います。論理的なフォーカス順序が重要です。

- - 画面拡大ソフトを使用する場合、視野が狭いため、フォーカス順序が論理的でないと誤った文脈でコンテンツの一部分を解釈してしまう恐れがあります。

事例

- 論理的で、使いやすいフォーカス順序は、ページの操作をキーボード使用に依存している運動障害のある利用者の役に立つ。

達成方法

WCAG ワーキンググループがこの達成基準を満たすのに十分であると判断する達成方法、又は複数の達成方法の組み合わせを表している。

しかしながら、必ずしもこれらの達成方法を用いる必要はない。

その他の達成方法についての詳細は、[WCAG 達成基準の達成方法を理解する](#)の「その他の達成方法」を参照。

十分な達成方法

1. [G59: コンテンツ内のシーケンスと関連性に応じた順序でインタラクティブ要素を配置する](#)
2. 次の達成方法の一つを用いて、コンテンツ内の順序及び関係性に従った順序でフォーカスを要素に与える:
 - [C27: DOM の順序を視覚的順序と一致させる](#)
 - [PDF3: PDF 文書で正しいタブ順序と読み上げ順序を確保する](#)
3. 次の達成方法の一つを用いて、ウェブページを動的に変化させる:
 - [SCR26: 動的なコンテンツを、Document Object Model の、そのトリガーとなる要素の直後に挿入する](#)
 - [SCR37: デバイス非依存な方法でカスタムダイアログを作成する](#)
 - [SCR27: Document Object Model を用いて、ページ上にある複数のセクションを並び替える](#)

失敗例

以下に挙げるものは、WCAG ワーキンググループが達成基準の失敗例とみなした、よくある間違いである。

- [F44: 達成基準 2.4.3 の失敗例 — 意味及び操作性を保持しないタブ順序を作成するために、tabindex 属性を使用している](#)
- [F85: 達成基準 2.4.3 の失敗例 — 連続するナビゲーション順序において、トリガーとなるコントロールに隣接していないダイアログ又はメニューを使用している](#)

2025年6月26日 Takuya Nakajima

達成基準 2.4.4: リンクの目的(コンテキスト内)

ソース: [達成基準 2.4.4 リンクの目的 \(コンテキスト内\)](#)

原文

それぞれのリンクの目的が、リンクのテキスト単独で、又はリンクのテキストとプログラムによる解釈が可能なリンクのコンテキストから判断できる。

ただし、リンクの目的がほとんどの利用者にとって曖昧な場合は除く。

ざっくりまとめ

リンクテキストはリンク先がなんなのかをそれ単体でわかるようになるべくしてねという話。

2.4.4は少し緩和条件があって、リンクテキスト単体ではわからなくても、リンクからフォーカスを移さずに理解することができればOKとされてる

同じ段落(あるいはリストやテーブルのセル)内のテキスト、またはそのリンクがテーブルにあった場合、その行や列のヘッダセルのテキストからわかるのであればそれをフォーカスを移動させずに読むことができるので許可するとされてる

他にもaria-describedbyとかで関連づけるなどの手段でもOK

- プログラムによる解釈で文脈を補う場合は、その文脈部分をリンクより手前に置いとくのが好ましい(人は基本的に前方で説明されてるだろうと推測するから)
- 2.4.2でサイトのタイトルにはユニークでページの目的を示すテキストを設定しろと書いてある。つまり、遷移先のページのタイトルをリンクテキストにすることはベストな対応であるといえる
- 同じリンク先が設定されてるリンクのリンクテキストは全部一緒にしとくとパッと判断できていいね (3.2.4の一貫した識別性のガイドラインにも適合できる)

その理由とメリット

- 関心のないリンクをスキップしやすくする (運動障害あると一回行って戻るのが結構大変なので重要になってくる)
- 同一リンク先のリンクテキストがゆれてると認知能力に制約あるユーザーが間違っって何回も押してしまう
- 支援技術ユーザーがリンクの文脈をさくっと理解しやすくなる

例外

コンテンツの性質上リンク先がユーザーに伏せられるケースもあるが、その時はそういうもんなのでOK

例えば、ダンジョン系のゲームがあったとして、ドア1ドア2ドア3のどれを開きますか？みたいな選択肢があったとして、その遷移先をはっきりさせちゃうとゲームとして破綻しちゃうよね？みたいなノリ

事例(原文)

- 以下リンクは遷移先が明確なのでOK
 - [中世の歴史](#)には大量の虐殺があった
 - 自動投票機のアイコンと「アイルランド政府の電子選挙委員会」というテキストが一つのリンクになっている
- 以下のリンクはフォーカス移動なしで文脈を読めるのでOK
 - あるウェブサイトには、ニュース記事のコレクションが含まれている。メインページでは、各記事の最初の部分が少し表示され、その後に「続きを読む」というリンクがある

ダメな例

- 隣接する段落内のリンク

...

<p>ANAでマイルを貯めるとお得に海外旅行ができます。</p>

<p>続きを読む...</p>

...

- テーブルレイアウト時の隣接セル(非ヘッダ)関連のリンク

...

<table>

<tr>

<td>あなたとJava</td>

</tr>

<tr>

<td>

今すぐダウンロード！

</td>

</tr>

</table>

...

感想

同一セクション内の見出しで文脈を補えればOKと思ってたけどこれはフォーカス移動せずに到達する手段なさそうなのでダメっぽいわね

達成基準 2.4.5: 複数の手段

ソース: [達成基準 2.4.5 複数の手段](#)

原文

ウェブページ式の中で、あるウェブページを見つける複数の手段が利用できる。

ただし、ウェブページが一連のプロセスの中の 1 ステップ又は結果である場合は除く。

ざっくりまとめ

利用者が自分等のやりやすい方法でコンテンツを見つけることができるようにそのページに辿り着く方法を複数用意する

その理由とメリット

- たとえばロービジョンのユーザーは拡大鏡やスクリーンリーダーを用いてドリルダウンして物件一覧に到達するよりフリーワード検索で一覧行く方が楽だよね
- 認知障害のある人は複数のページを読みつつ移動するよりサイト全体を見渡すことができるサイトマップみたいなのを好んだりすることが多い

例外

問い合わせステップのconfirmとかthanksページは一連のプロセスの中の1ステップ扱いになるので複数の到達手段は不要

事例(原文)

- 検索メカニズム
 - ある大手食品加工企業は、その企業の製品を使用して作成されたレシピを含むサイトを提供している。そのサイトは、特定の食材を使ったレシピを検索するための検索メカニズムを提供している。さらに、食品のいろいろなカテゴリを挙げたリストボックスを提供している。その企業のスープ製品から作られたレシピのリストを含むページに移動するには、利用者は検索エンジンに「スープ」を入力してもよく、又はリストボックスから「スープ」を選択してもよい。
- ウェブページ間のリンク
 - あるヘアサロンが、サービスを宣伝するためにウェブサイトを作成した。そのサイトは 5 ページしかない。各ウェブページには、ウェブページ間を順に前後へ移動できるリンクがある。さらに、各ウェブページには、その他のウェブページへ移動するリンクの一覧がある。

- コンテンツがプロセス又はタスクの結果である場合 - 送金確認
 - あるオンライン銀行サイトでは、ウェブを通じて口座間の送金ができる。口座の名義人が送金を完了する以外には、送金確認ページを見つける方法はない。
- コンテンツがプロセス又はタスクの結果である場合 - 検索エンジンの検索結果
 - ある検索エンジンが、利用者の入力に応じた検索結果を提供している。その検索プロセスを行う以外に、その検索結果ページを見つける方法はない。

感想

以下記述があるがG64は関係あるんか...？

十分なテクニック

- 次のテクニックのうち二つ以上を用いる：
 - [G125: Providing links to navigate to related Web pages](#)
 - [G64: Providing a Table of Contents](#)
 - [G63: Providing a site map](#)
 - [G161: Providing a search function to help users find content](#)
 - [G126: Providing a list of links to all other Web pages](#)
 - [G185: Linking to all of the pages on the site from the home page](#)

達成基準 2.4.6: 見出し及びラベル

ソース: [達成基準 2.4.6 見出し及びラベル](#)

原文

見出し及びラベルは、主題又は目的を説明している。

用語

ラベル

ページ内のコンポーネントを識別するために利用者に提示されているもの。

注記: [名前 \(name\)](#) は隠されていて、支援技術に対してだけ公開される場合がある一方で、ラベルはすべての利用者に提示される。多くの場合 (すべてではないが)、名前 (name) とラベルは同じである。

注記: ラベルという用語は、HTML における label 要素だけに限定されない。

ざっくりまとめ

見出しが明快で内容がわかるように書かれていれば利用者は自分の探してる情報を簡単に見つけることができる

ラベルが明快で内容がわかるように書かれていれば利用者はページ内のコンポーネントを識別が簡単になる

1Point advice

- ラベルや見出しは内容がわかるように書かれているにこしたことはないが、長々と書けというわけではない。短くてもコンテンツを見つけてナビゲートする手がかりとして有用ならそれで十分なこともある
- このガイドラインはマークアップ的な関連付けの話はしていない(別の項目でして)

その理由とメリット

内容がわかるように記述された見出しは読み取りに時間がかかるユーザーや記憶能力に制約のあるユーザーに役立つ。とくにそれぞれのセクションの内容を予測できるようになるとより良い

事例(原文)

- ニュースサイト
 - ニュースサイトのホームに、その時間帯のトップニュースの見出しが並んでいる。それぞれの見出しの下には、記事の冒頭の 35 ワードと、記事の全文へのリンクがある。そして、見出しはどれも、記事の主題を明快に伝えている。

- 上手な文章の書き方ガイド
 - 文章の書き方に関するガイドに、次のようなセクション見出しがある。上手な文章の書き方、無駄な語句を取り除く、不要な語句を見つけ出す、などである。これらのセクション見出しは明快かつ簡潔で、情報の構造が見出しの構造にも反映されている。
- 様々な記事で一貫した見出し
 - ウェブサイトには、ある学会の会議での論文が掲載されている。その学会の会議への投稿する論文は、次のような構成とすることが必須となっている: 要約、イントロダクション、(その論文に固有なその他のセクション)、結論、筆者略歴、用語集、そして参考文献。そうして、論文の独自性とセクション見出しの一貫性とのバランスをうまくとりながら、各ウェブページのタイトルはそのページにある論文をはっきりと特定している。
- 利用者に氏名を要求するフォーム
 - 利用者に氏名を要求するフォーム。姓と名を要求する二つのテキストフィールドから成る。最初のフィールドは姓、次のフィールドは名とラベルづけされる。

感想

関連リソースのニールセンの記事面白い

<https://www.nngroup.com/articles/how-users-read-on-the-web/>

<https://www.nngroup.com/articles/applying-writing-guidelines-web-pages/>

- 簡潔さ、スキャン可能性、客観性:
 - ユーザーが効率的に情報を処理できるように、テキストを短く、キーポイントを目立たせ、過剰な宣伝文句を排除すること。
- 効果の検証:
 - 改善されたウェブサイトは、元のサイトに比べて使用可能性が159%向上しました。これは、課題をこなす時間が短縮され、エラーが減り、ユーザーの満足度が上がったことに基づいています。
- 具体的な改善事例:
 - 元のページと比較して、書き直されたページは、単語数が元の54%となるように短縮され、セクションを再編成し、重要なキーワードには強調表示が加えられました。

達成基準 2.4.7: フォーカスの可視化

ソース: [達成基準 2.4.7 フォーカスの可視化](#)

原文

キーボード操作が可能なあらゆるユーザインタフェースには、フォーカスインジケータが見える操作モードがある。

操作モードという表現について...

最近focus-visibleみたいにキーボード操作の時のみフォーカスインジケータを出すみたいなの
の採用や実装が進んでいる

それを踏まえるとざっくり常に出すとは言いづらいので操作モードの時に見えてればいいという表現
になっている

ざっくりまとめ

フォーカスインジケータは操作モードの時はずっと出しておこうね
表示に時間制限などはつけてはならないよ

その理由とメリット

- 基本的にキーボード操作してる有視覚ユーザー全員に役立つ
- 注意力欠如・短期記憶の制約、または遂行機能における制限のある利用者がフォーカスを
発見するのに役立つ

事例(原文)

- ユーザインタフェースのコントロールがフォーカスを受け取ると、その周りに視覚的に認識で
きる枠線が表示される。
- テキストフィールドがフォーカスを受け取ると、縦の棒がテキストフィールド内に表示されて、
利用者がテキストを挿入入力できることを示す、又は、すべてのテキストがハイライト表示さ
れ、利用者がそのテキストを上書きできることを示す。

ダメな例

- outline: none;で消しちゃう
- フォーカスインジケータそっくりのスタイルを非アクティブな要素に指定してしまってどれがフォーカスインジケータがわからなくしてしまう
- フォーカスインジケータの判別が見辛くなるような太さ・色の境界線を指定してしまっている (outline-offsetとか使うといいね)

感想

事例の二つ目はフォーカスインジケータの話じゃないけどのってるのは目的が関連してるからなのかな～

操作モードって表現がついてるのきにしてなかったけどそういうことなんだな

2025年8月7日 秦那実

<https://jira.next-group.jp/wiki/pages/viewpage.action?pageId=475091263>

2025年8月21日 千葉穂乃美

<https://jira.next-group.jp/wiki/x/Yn9-H>

2025年6月26日 Takuya Nakajima

達成基準 2.5.1: ポインタのジェスチャ (レベル A)

ソース: [達成基準 2.5.1 ポインタのジェスチャ](#)

原文

マルチポイント又は軌跡ベースのジェスチャを使って操作する機能はすべて、軌跡ベースのジェスチャなしのシングルポインタで操作することができる。ただし、マルチポイント又は軌跡ベースのジェスチャが必要不可欠である場合は例外とする。

注記

この要件は、ポインタの動作を解釈するウェブコンテンツに適用される (ユーザエージェントや支援技術の操作に必要なアクションには適用されない)。

ざっくりまとめ

ポインターの軌跡ベースの操作にはフリックやスワイプ、ピンチイン、ピンチアウトなど色々ある
ユーザーはデバイスをどんな風に操作してるかわからない。例えばヘッドポインタを使っていたり音声操作してたり。 そんな中で難しいジェスチャや指をN本使うことを前提としたアプリケーションを作るのはよくないよねって話

なので軌跡を使わないクリック・ダブルクリック・タップ、長押しみたいに一点の接合点で操作が完結できるようにしよう

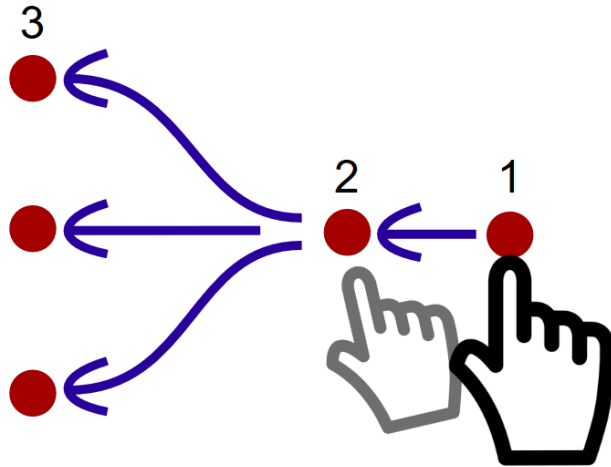
注: そのような機能を実装してはいけないというよりは、代替手段を用意してねという感じ

注: キーボードで操作できればいいのかという話ではない。 タッチデバイス等では物理キーボードを使わないユーザーもいるのでシングルポインタでかつ軌跡を利用しない代替手段が必要という話になる

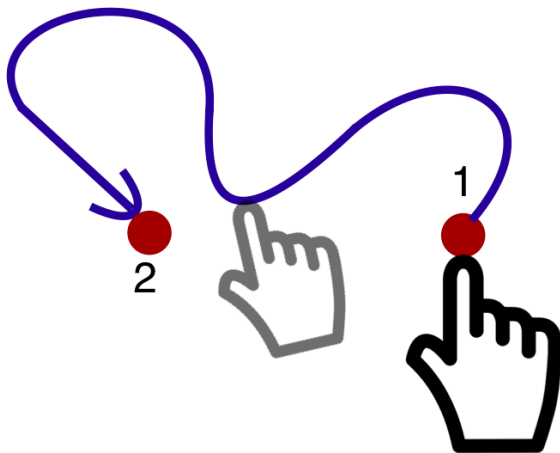
ちょっと細かい話

始点から終点まで水平に動かした時だけ認識されるようなケース. このケースでは3だと認識されないというパターン

これは軌跡ベースのジェスチャに含まれるのでアウト



始点から終点までの間に任意の一点を通過しないといけなようなパターン
これも軌跡ベースのジェスチャに含まれるのでアウト



Q. ドラッグは軌跡ベースなのか？

A. ドラッグは始点と終点の間に特定のポイントや制約を設けないので軌跡ベースではない。ただし別のガイドライン(2.5.7でドラッグによる操作はドラッグなしで代替できないといけない)に引がかかるのでOKというわけではない

事例

- 地図アプリケーションのズームをピンチ操作で行うことができるが、この要件を満たすため、ボタンも用意した

- 地図アプリケーションのズームをピンチ操作で行うことができるが、ダブルタップ→長押し→上で拡大、下で縮小といった方法でもできる
- ニュース記事をスワイプでページ送りできるが、ボタンでの操作もできる
- スライダーはタッチデバイスでは厳密な左右方向への移動ジェスチャが必要になるが、この要件を満たすため両サイドにボタンも用意しておいた

感想

キーボード操作できるようにする。に目がいきがちだけどタッチデバイスを考えるとなるほどシングルポインタ操作も保証しないといけないのねとなった。

とはいえキーボード操作対応したらボタン作ることになるだろうからそれによって満たせそうだなとも思った

これきっかけでヘッドポインタちょっと使ってみた

べろ出したらクリックとかしてみたらそれなりに使えたけどデュアルディスプレイだと徐々にズレが大きくなって大変だなと思った

達成基準 2.5.2: ポインタのキャンセル (レベル A)

ソース: [達成基準 2.5.1 ポインタのキャンセル](#)

原文

[シングルポインタ](#)を使って操作できる[機能](#)は、以下の要件の少なくとも 1 つを満たす。

- ダウンイベントがない
 - 機能を実行する目的でポインタの[ダウンイベント](#)を使用していない。
- 中止又は元に戻すことができる
 - 機能の完了には[アップイベント](#)を使用し、かつ機能の完了前に中止する、又は機能の完了後に元に戻すための[メカニズム](#)が利用できる。
- アップイベントで反転

- アップイベントによって、先のダウンイベントのすべての結果が反転する。
- 必要不可欠
 - ダウンイベントによって機能を完了させることが**必要不可欠**である。

注記

キーボード又はテンキーパッドのキープレスをエミュレートする機能は必要不可欠とみなされる。

注記

この要件は、ポインタの動作を解釈するウェブコンテンツに適用される (ユーザエージェントや支援技術の操作に必要なアクションには適用されない)。

ざっくりまとめ

Webページの送信ボタンとか、マウスで押し込んだあと、押し込んだままちょっとボタンから離してマウス離したら送信されないようにキャンセルのメカニズムがある

こういった機構をちゃんと用意することで誤った操作をしたときのリカバリになっていいねという話
とくに視覚障害、認知障害、運動障害等があるこういったミスは起きやすいので大変重要だという話

ざっくりした対応の話でいうとmousedownではなくmouseupやclickでハンドリングしようねという感じ。(mouseupやclickイベントはこのキャンセルメカニズムがそもそもついてるので離れたところでマウスアップされるとそもそも発火しない)

ドラッグアンドドロップの実装のようにmousedownしたタイミングで処理が始まる場合も規定の位置以外でmouseupが発生した時にキャンセルになるメカニズムがあると安心だね

ただし、射撃ゲームとかピアノアプリみたいに応答速度的にupだとちょっとずれちゃうよねって感じの時は必要不可欠とみなしてOKとする

事例(原文)

- 七並べをするときプレイヤーは手札をドラッグして場にカードを提出する。しかしドラッグ中に場以外のところでマウスアップするとカードは場には出されず手札に戻る

感想

- manager-neoのありとあらゆるボタンがmousedownで実装されてたな～、そもそもクリックイベントに紐づいてないからキーボード操作できないという致命的な問題があったがキャンセルできないって文脈もあったんだなとひしひし

達成基準 2.5.3: ラベルを含む名前(name) (レベル A)

ソース: [達成基準 2.5.3 ラベルを含む名前\(name\)](#)

原文

ユーザインタフェース コンポーネントがテキスト又は文字画像を含むラベルを持つ場合、視覚的に提示されたテキストが名前 (name) に含まれている。

注記

ベストプラクティスは、ラベルのテキストを名前 (name) の最初に使用することである。

ざっくりまとめ

- 可視ラベル(画面に見えるテキスト)を持つUIコンポーネントは、そのテキストをアクセシブルネーム(name)にも含めること。
- 可視ラベルがないコンポーネントにはこのガイドラインは適用されない。プレースホルダーは基本ラベルの代替ではないが、アクセシブルネームの算出ソースの最低優先度として組み込まれて入る
- アクセシブルネームは様々な方法で設定することができる。 aria-label/aria-labelledbyはその中でも採用優先度の高いネーミング手法なので他の可視ラベルを上書きすることがある。 注意して使う必要がある。
- 記号的な文字やアイコンをラベルとして使っているコンポーネントの場合(例えば、斜体にするボタンが i とかかかれていたりするケース)、それは有効なラベルとはみなされずこのガイドラインの範疇ではない(文字画像に言及しているWCAG 1.4.5などが対象となる)

- 名前に「name :」のようにコロンが含まれていても別に問題はない。音声入力のケースにおいても記号部分は発話しなくていいので影響がない。
- ラベルが数式だった場合(例: $11 \times 3 = 33$)、それはラベルとして有効であり、上書きする必要はない。

その理由とメリット

- 音声入力ユーザーは画面に見える語をそのままコマンドとして話す。可視ラベルとnameが一致していれば、ボタンやリンクに迷いなくアクセスできる
- スクリーンリーダー利用者も弱視ユーザー等の場合、聞こえるテキストが見えているテキストと一致することで認知負荷が下がる

ダメな例

- アクセシブルネームが可視ラベルのテキストを含まない
 - 例: 可視ラベルが「Go」なのに `aria-label="キーワードで探す"` になっている
 - 例: 可視ラベルが「Delete」なのに `aria-label="Remove item"` になっている(“Delete” が name に含まれない)
- プレースホルダーをラベルとして使用する。

感想

意味合いがあつてれば名前の上書きは別にいいんだとおもってたけど、音声操作ユーザーの誤操作考えたらダメそうだった

あと現段階ではピックアップされてないが、可視ラベルのテキストがアクセシブルネームに含まれる場合、その位置にも制約ができそう(最前方じゃないと音声操作時にあやまったところにフォーカスいく可能性かまるからダメって話だと思う)

達成基準 2.5.4: 動きによる起動 (レベル A)

ソース: [達成基準 2.5.4 動きによる起動](#)

原文

ざっくりまとめ

動き(端末の傾き・振り)で操作できる機能は、同じ機能を必ず通常のUIコンポーネントでも操作できるようにすること

また、動きによる操作は無効化できる手段を提供すること。(ボタンでもいいし、OSの設定で関連するものがあればその設定に応じて無効化できるとかでもいい)

例外は2つだけ:

- 支援技術だったり、ブラウザのアクセシビリティ機能だったり提供してるもの
- 機能にとって動きが必要不可欠な場合

必要不可欠とみなされる対象は「位置情報やビーコンで検出される移動」、「キーボードとかマウスの操作に付随する動き等」など

その理由とメリット

- 端末を安定して持てない・動かせない人にも操作手段を担保できる
- 神経の病気由来の震えなどにより意図せず起動する誤作動を防止するため

事例(原文)

- 利用者は、シェイクして元に戻す機能及びその他の動きによる起動の機能をオフにするアプリケーション設定を選択できる
- テキストを入力フィールドに入力した後、デバイスを振るとダイアログが表示され、入力を元に戻すことができる。テキストフィールドの隣にあるキャンセルボタンも同じ機能を提供する。

- 利用者はデバイスを傾けて次、又は前のページに進むことができる。ボタンも同じ機能を実行するために提供されている。

感想

jsのmatchMediaとかでた当初は使い道、レスポンシブ対応系以外で全然イメージできてなかったけどこの辺の対応とかをOSの設定に合わせたいとかいうときモリモリ使うんだらうな～って10年越しくらいに思ったりした

達成基準 2.5.5: ターゲットのサイズ (高度) (レベル AAA)

ソース: [達成基準 2.5.5 ターゲットのサイズ\(高度\)](#)

原文

ポインタ入力のターゲットのサイズは、少なくとも 44 × 44 CSS ピクセルである。ただし、次の場合は除く:

同等

ターゲットと同等のリンク又はコントロールが同じページに 44 × 44 CSS ピクセル以上のサイズで存在する。

インライン

ターゲットが文中、又はテキストブロック内に存在する。

ユーザエージェントのコントロール

ターゲットのサイズがユーザエージェントによって決定され、かつコンテンツ制作者によって変更されていない。

必要不可欠

そのターゲットを特定の方法で提示することが、伝達される情報において必要不可欠である。

ざっくりまとめ

まずターゲットとは...

UIのインタラクティブな領域(ボタンとか)のこと

ターゲットは大っきいほうがいいよ、具体的には44px以上だね。 そうしとくと特にタッチデバイスでは操作ミスが減っていいよ

ポインティングデバイスを使うユーザーだったとしてもそれがアイトラッキングとかだったり、運動障害をもったりすると精密な操作はできない、そういった時にターゲットが大きいと助かる

大きいとロービジョンのユーザーを見つけやすくなるといった点も大きいね

このガイドラインで提示してる44pxは最低限のサイズなので頻繁に使われるコントロールだったり、後戻りできないアクションを伴うボタンだったり、画面の端っこの遠いところにあったり、一連の流れの中で操作されるUIだったりしたときはもっと大きくしとくといいね★

感想

原文のところでインラインが許可されている。 つまり文中リンクは許可されるんだと思う。

[abemaのガイドライン](#)みてリンクも対象だという認識してたが”文中の”リンクは許可なんだな〜と悟った

一方これは文中ではないので多分アウト



これ緩和基準のAAがあると思うんだけどみたら2.5.8で珍しく厳しいガイドラインの方が前方にあった時系列みたらAAA基準がwcag2.1で登場して、wcag2.2で緩和基準がでてくるってことらしい珍しいね

ずっと思ってることとしてカルーセルのスライドピッカーコントロールってターゲットの最小下回るのどう扱うのがいいんだろう

2025年11月27日 千葉穂乃美

https://jira.next-group.jp/wiki/x/a_kwHQ

2025年12月11日 Takuya Nakajima

達成基準 3.1.1: ページの言語(レベルA)

原文:

それぞれのウェブページのデフォルトの自然言語 がどの言語であるか、プログラムによる解釈が可能である。

ざっくりいうと...

htmlタグのlang属性にしっかりそのページで利用してる主言語を指定しようという話

なんで？

スクリーンリーダー文脈だと...

htmlのlang属性をみて音声エンジンを切り替えてるので、日本語のページにenが設定されていた場合、正しく発音されなかったり、読み上げ自体が行われなくなったりなど壊滅的な被害が発生する

<https://x.com/miyabin4113/status/1964157737400676453>

<https://x.com/cloud10designs/status/1396817592455634952>

<https://x.com/ymrl/status/1377928280435683332>

https://x.com/o_mergen7/status/1300093782499573760

<https://x.com/masuP9/status/1246707814170411008>

https://x.com/ME_TKS_/status/1235167003469479936

https://x.com/robert_KIMATA/status/1021595554386608128

他にもページ翻訳で元言語の抽出に使われるものもある(google translateはみてないってってるけど)

よくある失敗ケース

- frameworkの雛形作成コマンドを作って生成したhtmlのlangの指定がないままであったり、英語圏で作られたものであればenが指定されていることが多く、それをそのまま使ってしまう

達成基準 3.1.2: 一部分の言語(レベルAA)

原文:

コンテンツの一節、又は語句それぞれの自然言語がどの言語であるか、プログラムによる解釈が可能である。ただし、固有名詞、技術用語、言語が不明な語句、及びすぐ前後にあるテキストの言語の一部になっている単語又は語句は除く。

ざっくりいうと...

サイト内でページの主言語以外の言語が使われている場所があれば、そこに個別のlang属性を設定してください

なんで？

3.1.1と一緒にlang属性を指定しておくとその領域の読み上げの音声エンジンがその言語特有のものに切り替わって適切に読み上げてくれる(イントネーション含め)

左から右に流れる日本語のような言語を主言語としたページ内に右から左に流れるようなアラビア語テキストがあったときとかのレンダリングを適切に行うみたいなことにも貢献するケースがあるらしい

(これに関してはモダンブラウザはすべて関係なくレンダリングに成功したので以前は...という話っぽい)

例外はあるの？

- 固有名詞はいいよ。人名とか製品名とかしょうがないよね

- 技術用語とかもいいよ。(インターネットとかヘルツとか)
- 言語が不明な語句もいいよ
- 前後のテキストの言語の一部になってる単語や語句もいいよ

単語や語句によってはその言語の単語ではなくてもその言語で使われる単語がある

```
<p lang="en">I had pizza and sushi for dinner.</p>
```

のpizzaはイタリア語、sushiは日本語由来だが、英語話者は両方とも英語の発音体系でそのまま読めるし、辞書にもものっている。

こういったものは合成エンジン側もenのエンジンで適切に発音できるので問題はない

一つの文章中にある一連の言語の中に混じる別言語が、自然な感じで組み込まれているなら、それは一般的な事象なので主言語の一部として捉えていい

例:「年末調整のWeb申告の期限は明日までです」

よくある失敗ケース

- LQSのサイトの各項目にlang属性を設定してないため、Focusedがフォキューズドとよまれる

達成基準 3.1.3: 一般的ではない用語(レベルAAA)

原文:

慣用句及び業界用語を含めて、一般的ではない、又は限定された用法で使われている単語、又は語句の、明確な定義を特定するメカニズムが利用できる。

ざっくりいうと...

難しい言葉や一般的でない言葉には説明をいれるか、用語集みたいなものへのリンクを用意しましょう

なんで？

文字通りの意味じゃない言い回し(慣用表現)とか難しい言葉を苦手とする障害がある人の理解が困難になる可能性がある。

また、その言語が母国語じゃない人間にとっても難しいよね

中学英語を学んだ程度の英語話者に、eat crow(カラスを食べる)が「自分の誤りを認める」とか「屈辱を味わう」みたいな意味になることを分かれというのは無理があるよね

どんな方法があるの？

- 用語集を用意してそこにリンクをはる (LHは結構このパターンを使う)
- 単語のとなりに()で説明を入れる
- 単語のとなりに*1みたいなページ内リンクを入れておいてページ末尾に対応する説明をいれておく

よくある失敗ケース

- ???「うちは今期、売上の伸びよりも**EBITDA**マージンの改善を優先してきました。理由は単純で、「規模の幻影」より「利益の現実」の方が、よほど会社を強くするからです。」

LIFULLがパートナー向けに共有してるコンテンツ執筆ガイドライン

https://docs.google.com/document/d/1yYgkHCiqJn_X0QuugH_KFO_f5bWt4eiNwImIaVkbkW8/edit?tab=t.0