

JIS 実装チェックリストひな型作成について

2010/01/27 NTT 渡辺昌洋

目的

実装チェックリストの作成方法が不明であるため、作成方法を明らかにする

作業内容

JIS 実装チェックリスト関連文書群の作成

1. 実装チェックリスト作成手順
2. 実装チェックリスト作成ルール
3. 実装チェックリストひな型
4. 実装チェックリスト記入ガイド

公開予定

2010 年 ? 月 JIS 改定に合わせ CIAJ ウェブサイトで公開

作業現状

実装チェックリスト作成手順、作成ルール、ひな型を作成済み

今後のスケジュール案

2 月 1 日 - 2 月 28 日

達成基準 1.1.1 に対応する実装チェックリスト作成トライアル

3 月 1 日 - 3 月 31 日

トライアル結果検証、問題点の洗い出し

4 月 1 日以後

達成基準 1.1.1 以外に対応する実装チェックリスト作成トライアル

トライアル結果検証、問題点の洗い出し

5 月末 版公開

達成基準 1.1.1 に対応する実装チェックリスト作成トライアル

作業概要

- 「実装チェックリスト作成手順案」の作成手順に従って、複数の作業者が、同じ実装チェックリストを作成する。できあがった実装チェックリストと、リスト作成時の意見などから、問題点を明らかにし、作成手順をブラッシュアップする。

実装チェックリスト作成対象

- 使用する実装方法が不明の場合について、実装チェックリストを作成する（つまり、すべての実装方法の組み合わせについて実装チェックリストを作成する）。
- 達成基準 1.1.1 のみを対象とする。
 - 分担するのではなく、同じ対象について、実装チェックリストを作成することにより、問題点をより多く発見する。

作業メンバ

- 各委員にご担当いただきたい。
-

実装チェックリスト作成手順案

1. 適用する達成基準に対して、Understanding WCAG 2.0 翻訳の「達成基準を満たすことのできる実装方法」を参照して、必要な実装方法（の組み合わせ）を確認する
2. Techniques for WCAG 2.0 翻訳を参照し、必要な実装方法（の組み合わせ）に対応する試験方法(tests)の中で、満たすべき項目(expected results)を確認する
3. 満たすべき項目(expected results)からチェック項目を作成する
 - 満たすべき項目ごとにチェック項目を作成する
 - 関連する実装方法が複数ある場合には、Understanding WCAG 2.0 翻訳の「達成基準を満たすことのできる実装方法」を参照して、なるべく重複をなくすように、チェック項目を作成する
 - チェック項目をより具体的に書ける場合は、具体的に書いて、チェック項目としてもよい
4. まとめた項目を実装チェックリストの項目として記載する
 - 対応する解説書項目の欄には、Understanding WCAG 2.0 翻訳のどの項目を参照したかがわかるように、状況と項目番号を記載する
 - 例えば、Understanding WCAG 2.0 翻訳における記載が、状況 A の 2 番目の項目であれば、状況 A-2 と記載する

（注）ここでは、実装チェックリストの各行を、チェック項目と呼ぶことにする。チェック項目のリストが実装チェックリストとなる。

事例：「input 要素を使って、画像で送信ボタンを実装する」際のチェック項目の作成

```
<input type= " image " alt= " 送信ボタン" >
```

達成基準 1.1.1 の「達成基準を満たすことのできる実装方法」を [Understanding WCAG 2.0 翻訳](#)で調べる

Understanding WCAG 2.0 翻訳の記述（文末資料参照）

- ◆ 達成基準 1.1.1 の実装方法及び不適合事例 - 非テキストコンテンツ
- ◆ 達成基準を満たすことのできる実装方法
 - 状況 A： ...
 - 状況 B： ...
 - 状況 C: 非テキストコンテンツがコントロールである、又は利用者の入力を受け入れる場合：
 - 次に挙げる短い代替テキストの実装方法を用いて、G82：非テキストコンテンツの目的を示す代替テキストを提供する
 - ...
 - 前述の "達成基準を満たすことのできる実装方法" を用いる際の短い代替テキストの実装方法
 - H36：送信 / 実行ボタンとして用いる画像の alt 属性を使用する (HTML)
 - ...

Techniques for WCAG 2.0 翻訳の記述

- G82：非テキストコンテンツの目的を示す代替テキストを提供する
- 試験
 - 手順
 1. 非テキストコンテンツを、除いたり、隠したり、マスクしたりする
 2. 非テキストコンテンツを代替テキストに置き換える
 3. 機能が失われても非テキストコンテンツの目的が明らかかをチェックする
 - 期待される結果
 1. #3 を満たしていること。

- H36：送信 / 実行ボタンとして用いる画像の alt 属性を使用する (HTML)
- 試験
 - 手順
 1. type 属性値が image である、すべての input 要素において、alt 属性があるかチェックする
 2. alt 属性がボタン機能を示しているかチェックする
 - 期待される結果
 1. #1 と#2 を満たしていること

満たすべき試験項目

- H36 (手順 1)： type 属性値が image である input 要素に alt 属性があること
- H36 (手順 2)： type 属性値が image である input 要素の alt 属性がボタン機能を示していること
 - H36 (手順 2) が満たされると、G82 (機能が失われても非テキストコンテンツの目的が明らかであること) も満たされることになるので、H36 (手順 2) にまとめた

実装チェックリストに記載するチェック項目

- type 属性値が image である input 要素に alt 属性があること (1.1.1 状況 C-1)
- type 属性値が image である input 要素の alt 属性がボタン機能を示していること (1.1.1 状況 C-1)

作成される実装チェックリストの例

達成基準		対応する	等級	適合	試験方法	備考
項番	実装方法	Understanding WCAG 2.0				
	7.1.1.1 非テキストコンテンツ		A			
1	type 属性値が image である input 要素に alt 属性があること	1.1.1 状況 C-1				
2	type 属性値が image である input 要素の alt 属性がボタン機能を示していること	1.1.1 状況 C-1				
3	:					
	(以下省略)					

(資料) **Understanding WCAG 2.0 翻訳抜粋 達成基準 1.1.1**

達成基準を満たすことのできる実装方法

使用法： そのコンテンツに合致する状況を以下から選択すること。それぞれの状況には、WCAG ワーキンググループがその状況において十分であると判断する、番号付の実装方法（又は、実装方法の組合せ）がある。

状況 A： 短い説明によって、非テキストコンテンツと同じ目的を果たし、同じ情報を提示できる場合：

1. 次に挙げる短い代替テキストの実装方法を用いて、G94：非テキストコンテンツと同じ目的を果たし、同じ情報を提示する短い代替テキストを提供する

状況 B： 短い説明によって、非テキストコンテンツと同じ目的を果たし、同じ情報を提示できない場合（例：チャート又はダイアグラム）：

1. 次に挙げる短い代替テキストの実装方法及び次に挙げる長い説明の実装方法の一つを用いて、G95：非テキストコンテンツを簡潔に説明する短い代替テキストを提供する：
 - 次に挙げる長い代替テキストの実装方法を用いて、G92：非テキストコンテンツと同じ目的を果たし、同じ情報を提示する長い説明を提供する
 - G74：短い説明の中で長い説明のある場所を示して、非テキストコンテンツの近くにあるテキストで長い説明を提供する
 - G73：非テキストコンテンツのすぐ隣に別の場所へのリンクを置き、その別の場所で長い説明を提供する

状況 C： 非テキストコンテンツがコントロールである、又は利用者の入力を受け入れる場合：

1. 次に挙げる短い代替テキストの実装方法を用いて、G82：非テキストコンテンツの目的を示す代替テキストを提供する
2. H44：label 要素を用いて、テキストのラベルとフォーム・コントロールを関連付ける (HTML)
3. H65：label 要素を用いることができないとき、title 属性を用いてフォーム・コントロールを特定する (HTML)

状況 D： 非テキストコンテンツが時間の経過に伴って変化するメディアである場合（ライブの映像しか含まないコンテンツ及びライブの音声しか含まないコンテンツを含む）；テキストで提示されると無効になる試験又は演習；又は、特定の感覚的体験を創り出すことを主に意図しているコンテンツ：

1. 次に挙げる短い代替テキストの実装方法を用いて、ラベルを記述する。

- 次に挙げる短い代替テキストの実装方法を用いて、[G68: ラベルを記述して、ライブの音声しか含まないコンテンツ及びライブの映像しか含まないコンテンツの目的を説明する](#)
- 次に挙げる短い代替テキストの実装方法を用いて、[G100: 非テキストコンテンツの一般的な名前、又は説明的な名前を提供する](#)

状況 E: 非テキストコンテンツが CAPTCHA である場合:

- [G143: 代替テキストを提供して、CAPTCHA の目的を説明する、かつ G144: 異なるモダリティを用いて、同じ目的を果たす CAPTCHA をもう一つウェブページで提供する](#)

状況 F: 非テキストコンテンツを支援技術が無視するようにしなければならない場合:

- 次に挙げるウェブコンテンツ技術特有の実装方法を用いて、支援技術が非テキストコンテンツを無視するように実装する、又はマークアップする。
 - [H67: 支援技術が無視すべき画像の img 要素で、alt 属性値を空にして、title 属性を付与しない](#) (HTML)
 - [C9: CSS を用いて、装飾的な画像を付加する](#) (CSS)

前述の「達成基準を満たすことのできる実装方法」を用いる際の短い代替テキストの実装方法

- [H36: 送信 / 実行ボタンとして用いる画像の alt 属性を使用する](#) (HTML)
- [H2: 隣り合った画像とテキストリンクを同じリンクの中に入れる](#) (HTML)
- [H37: img 要素の alt 属性を用いる](#) (HTML)
- [H35: applet 要素に代替テキストを提供する](#) (HTML)
- [H53: object 要素のボディに代替テキストを記述する](#) (HTML)
- [H24: イメージマップの area 要素に代替テキストを提供する](#) (HTML)
- [H86: ASCII アート、絵文字、及びリート語に代替テキストを提供する](#) (HTML)
- [H30: a 要素のリンクの目的を説明するテキストリンクを提供する](#) (HTML)

注記: 「[達成基準 2.4.4 文脈におけるリンクの目的](#)」を理解するを参照。

- [G196: 画像のグループにある一つの画像に代替テキストを提供して、そのグループのすべての画像を説明する](#)

前述の「達成基準を満たすことのできる実装方法」を用いる際の長い代替テキストの実装方法

- [H45: longdesc 属性を用いる](#) (HTML)
- [H53: object 要素のボディに代替テキストを記述する](#) (HTML)