

Accessibility on Web

PV219, spring 2023

Agenda

- Motivation / Why We Care?
- Disabilities
- Misconceptions
- WAI-ARIA
- WCAG / Legal Requirements
- Making Accessible Content
- Tooling and Testing
- Demonstration
- Summary

Motivation / Why We Care?

- **15 %** of the global population report challenges in dealing with basic daily tasks and interactions ([WHO](#))
- **>30 %** of us will have some form of disability by the time we retire ([StatsCan](#))
- We all benefit from **clear, logical, and user-friendly content** and experiences
- Accessible websites use **semantic HTML** and can help **increase organic search traffic** by as much as **50 %**

Disabilities

There are three modes of disability:

- 1. Permanent Disability:** This is when you are completely disabled. Example: blind, deaf and so.
- 2. Temporary Disability:** This is simply a physical or mental disability which hinders your discharging of responsibilities for a short period of time.
- 3. Conditional or Situation Disability:** This is simply when you are not able to do things due to the situation you find yourself in. Example: slow internet connection.

Disabilities

Category	Examples of Disabilities	Examples of Needs
Visual	Blind, low vision, color blindness	Screen reader, keyboard navigation, enlarged content, high contrast
Audial	Deaf, hearing impaired	Video captions, audio transcripts
Mobility	Limited motion, coordination	Keyboard navigation, large controls
Cognitive	Learning, memory, dyslexia	Simple content, layout, navigation

Assistive Technologies

Assistive technologies help users with disabilities increase, maintain, or improve their functional capabilities.

- Screen readers
- Eye tracking software
- Keyboard shortcuts
- Braille displays

Misconceptions in A11Y (1/3)

“Web accessibility only helps a fraction of my users / the population.”

- For instance, 3.8 million Canadians over 15 years old (**14 %**) live with disabilities

Big picture: There are significant overlaps between **accessibility and usability** that can benefit everyone.

Misconceptions in A11Y (2/3)

“Web accessibility is expensive and time consuming.”

- More research is required to create accessible sites, but designers and developers experienced in accessibility require less research time and bake these “extra features” in from the get-go
 - [Retrofitting inaccessible sites](#) is a lot more time consuming (and funds) than building an accessible site from scratch

See [W3C’s Financial Factors of web accessibility](#) for more details.

Misconceptions in A11Y (3/3)

“We don’t need to be compliant.”

- Legislation (like the Czechia's Disabilities Act) is [still evolving](#)
- Web accessibility lawsuits are steadily increasing in the U.S.
 - The precedent has been set: a blind plaintiff unable to purchase products online has [successfully sued Winn-Dixie](#), a major American grocery chain

WAI-ARIA

Web Accessibility Initiative – Accessible Rich Internet Applications is a technical specification published by the W3C that specifies how to increase the accessibility of web pages, in particular, dynamic content, and user interface components developed with Ajax, HTML, JavaScript, and related technologies.

[Some practical tips](#) are available in special document on W3C's site.

WCAG

The ***Web Content Accessibility Guidelines*** are a series of [internationally recognized](#) (available also in [Czech language](#)) guidelines that are intended to help solve many problems that web users with disabilities face. Guiding principles are:

- **P**erceivable
- **O**perable
- **U**nderstandable
- **R**obust

Three levels of conformance: A, AA, AAA

WCAG

Perceivability is converting non-textual content into text, which can then be processed by the assistive technology of the user's choice. **Think about:** Alt text, Captions

- **Orientation:** Websites and applications must support both portrait and landscape modes, unless absolutely necessary. This allows users with visual disabilities to rotate their mobile devices to increase text size.
- **Identify Input Purpose:** Websites must include indications about what kind of data to enter in a field. This allows the browser to autofill some forms, and allows assistive technologies to better inform the user about the purpose of different fields.
- **Identify Purpose:** Similarly, interface components such as icons should have specific labels within the code that assistive technologies can interpret. For example, a button that returns the user to the main page should be labeled “home button.”
- **Reflow:** Websites must use responsive design so that the text can be enlarged while the layout is preserved.
- **Non-Text Contrast:** To assist users with low vision, active interface components and non-text content should have a contrast ratio of at least 3:1.
- **Text Spacing:** Users must be able to increase the spacing between lines, paragraphs ,and words without losing functionality.
- **Content on Hover or Focus:** A user should be able to be dismiss pop-up content that appears in a modal window or tooltip at will and without having to move the pointer hover or the keyboard focus.

WCAG

Operability means to allow users to navigate focusable elements via the keyboard and activate them using the *Return* key. Make sure that you do not override browsers behavior or use an illogical focus order. **Think about:** Focus states, Keyboard navigation, Form labels, Avoid flashes or fast animations

- **Character Key Shortcuts:** Keyboard shortcuts that use a letter, punctuation, number, or symbol can be turned off or changed to use keys such as Ctrl and Alt instead.
- **Timeouts:** Websites must inform users of how much time they have before inactivity leads to data loss, unless the time limit is longer than 20 hours.
- **Animation from Interactions:** Users can turn off animations and videos, unless they are essential to the website's functionality.
- **Pointer Gestures:** Users can replace complex gestures such as pinching and swiping with simpler gestures such as taps and long presses.
- **Pointer Cancellation:** Users must be able to easily cancel an accidental "down event" such as a click, tap, or long press.
- **Label in Name:** A label's visible text and accessible name must match each other.
- **Motion Actuation:** Functionality that is accessible through motions such as shaking your mobile device must also be accessible through the user interface.
- **Target Size:** In most cases, clickable elements must be at least 44x44 pixels.
- **Concurrent Input Mechanisms:** Users should be able to switch between multiple input mechanisms, such as touch, keyboard, mouse, and speech.

WCAG

Content that is **understandable** can be read and comprehended by users without undue effort. This means that the content should be understandable both by the users themselves and by assistive technologies such as screen readers. **Think about:** Link text, Form errors

- **Readable:** Some people with disabilities experience challenges when recognizing written words or when inferring the meaning of a word from context. Your web content should make this process as easy as possible by identifying the language(s) that text is written in as well as any unusual words or abbreviations.
- **Predictable:** Presenting your website content in a predictable order and having your site behave predictably are essential for people with disabilities to successfully use and navigate your site.
- **Input Assistance:** People with disabilities, and the assistive technologies they use, often make mistakes more easily when entering information. In order for them to understand the problem and correct the issue, websites should provide better error messages and help prevent errors whenever possible.

WCAG

The **robustness** can be satisfied automatically if you use semantic elements correctly for their intended purpose. Maximize compatibility with current and future user tools.

Think about: Semantics, Valid HTML

- **Parsing:** The content and code of your website should be well-formed. For example, content written in a markup language such as HTML or XML should have complete start and end tags and should nest elements correctly. This will help prevent display errors and problems with assistive technologies.
- **Name, Role, and Value:** User interface components such as form elements can have their name and role “programmatically determined” by an assistive technology. In addition, people with disabilities can use assistive technologies to set values, properties, and states on your website.
- **Status Messages:** Many websites make use of dynamic content such as status messages that are written in markup languages such as HTML and XML. This content must be presented to users of assistive technologies without necessarily receiving a visual focus. For example, if users are viewing their social media feeds, they can be alerted of a new post without the browser automatically scrolling up to display it to them

WCAG

Level A

- High user impact
- Low impact on presentation/functionality
- Easy to implement

Captions are provided for all pre-recorded videos with audio

WCAG

Level AA

- High user impact
- Some impact on presentation/functionality

An audio description is provided for all pre-recorded videos with audio.

WCAG

Level AAA

- Specific user impact
- Some impact on presentation/functionality
- Increased level of difficulty to implement

Sign language interpretation is provided for all pre-recorded videos with audio.

Legal Requirements

- **WCAG 2.0** is the **de facto standard**
- Countries whose gov't sites must comply with at least WCAG 2.0 **Level A**:
 - Japan
 - New Zealand
 - **All of the EU**
- Countries with broad accessibility laws that have been applied to the web
 - USA (Lawsuits interpreting this as meeting WCAG Level A/AA)
 - Australia (Gov't interpreting this as meeting WCAG Level A)
 - UK (Discrimination lawsuits settled out of court; Gov't interpreting this as WCAG 2.0)

Making Accessible Content

Colors

The **red-green color deficiency**, affects approximately 8 % of the population. On the other hand users with **learning disabilities**, benefit greatly from color when used to distinguish and organize your content.

- Be sure to use other visual indicators than colors (asterisk, question mark)
- Use visual separation between block (whitespace, borders)

Making Accessible Content

Navigation

- Navigation should be simple and consistent
- Navigation links should come after the main content
 - Or provide a link or method that allows users to skip repetitive navigation links
- Display some content only to screen readers
 - Use absolute positioning off the screen instead of `display:none` or `visibility:hidden`
- Links should describe the linked page and make sense if read out of context
 - Avoid *"click here"* or *"more"*
- Buttons and tabs should be large enough for easy use

Making Accessible Content

Tables

- Data tables: `summary="..."`, `<caption>`, `<th>`, `scope="col"`
- Layout tables: `role="presentation"`
- Keep tables simple; minimize nested tables and cells spanning rows and column

Making Accessible Content

Online Forms

- **Easy** to navigate, complete, and submit
 - **Instructions** positioned before the form elements; clearly identify required fields
 - **Keyboard** for navigation (e.g. Tab and arrows to move, Enter to submit) and logical tab order
 - **Label** form fields (e.g. text, dropdowns): `<label for="name">Name:</label><input type="text" id="name" name="name" />`
 - **Group** related fields with `<fieldset>` and `<legend>` tags
 - **CAPTCHA** alternatives
- **Error handling**
 - **Validation**: both client and server
 - Error **messages** that clearly indicate the problem and propose a solution
 - Error **correction** should be easy
 - **Alternative** way of providing the information requested, e.g. e-mail

Making Accessible Content

Inclusive Disabled Buttons

- 01 – On hover, change the cursor to `cursor: not-allowed`.
- 02 – Always keep disabled buttons *focusable*.
- 03 – On focus/tap/click, explain why the button is disabled.
- 04 – We can show a *tooltip* or a text message on focus/tap/click.
- 05 – To avoid double bookings, add a spinner/change the wording on tap/click.
- 06 – Prevent double click programmatically via JS with `aria-disabled`.
- 07 – Use *ARIA live regions* to announce dynamic content.
- 08 – Avoid `pointer-events: none` as it won't prevent focus/key nav.
- 09 – Guide the user to error summary, or to form errors directly.
- 10 – Provide a “*way-out*”-link under the disabled button.
- 11 – Allow customers to overrule errors and continue despite them.
- 12 – Keep the “Continue” button accessible and validate on submit.
- 13 – Consider replacing disabled buttons with more actionable alternatives.

Making Accessible Content

Language

Indicating the language of content using the `lang` attribute on multilingual sites ensures assistive technology will read content with correct pronunciation.

- Add a global language on the `<html>` tag.
- Add a `lang` attribute to any in page content that differs in language.

Tooling and Testing

A few accessibility tests you can give yourself immediately:

- Don't use a mouse
- Zoom your page to 200 %
- Use your site on a phone
- Turn on high contrast

Free accessibility tools and assistive technology you can use: [NVDA](#), [TalkBack](#), [Seeing AI](#), [Google A11Y Tools](#), ...

Tooling and Testing

Automated Tools

Accessibility Evaluation Tools ([WAVE](#) and [aXe](#)) analyze website markup against various WCAG criteria and run right in your browser.

These tools help to detect:

- Missed heading levels
- Empty links and alt text
- Color contrast issues

Tooling and Testing

The following apply to the entire page:

- h1

Summary

14 Errors	2 Contrast Errors
37 Alerts	14 Features
40 Structural Elements	14 ARIA

View details >

Without admission exams | 25 years of FI | Feel FI | Thesis with industry | Office for studies

```
<html class="js touch" lang="en">
<head>
  <base href="https://www.fi.muni.cz/">
  <!-- [if lte IE 9] <html lang="en" class="old-browser no-js no-touch"> <![endif]-->
  <!-- [if IE 9 ] <html lang="en" class="ie9 no-js no-touch"> <![endif]-->
  <!-- [if (gt IE 9)|!(IE)]><!-->
  <!-- <![endif]-->
  <!-- ** HEAD ****>
  <title>
    Faculty of Informatics, Masaryk University | Faculty of Informatics Masaryk University
```

Tooling and Testing

Vision simulators

See what users see by running vision simulators in the browser. These programs ([Color Contrast Analyzer](#), [NoCoffee](#)) help identify areas that may be failing WCAG contrast ratios or are generally problematic.

Tooling and Testing

The image shows a screenshot of the eBay website with several accessibility testing annotations. The top navigation bar includes links for 'Life', 'Deals', 'Gift Cards', 'Sell', and 'Help & Contact'. A 'SUN-SOAKED SAVINGS' banner is circled in pink. The main content area features a large promotional banner with the text 'THIS IS MORE THAN SUMMER SHOPPING' and 'THIS IS MAKING MEMORIES TO LAST A LIFETIME', both circled in pink. Below this is a 'Find what inspires you' button, also circled in pink. On the right, there is a 'Zip Code' input field and a 'Get a Quote' button, both circled in pink. The bottom right corner contains the text 'LET EBAY VAULT SELL'.

Hide Mask WCAG Level Evaluated: Level AA, Small Non-Bold Text (4.5:1) Pixel Radius: 1 Rescan Download

Status: Complete

Life | Deals | Gift Cards | Sell | Help & Contact SUN-SOAKED SAVINGS

ebay All Categories Search

Following This Latest Popular Easiest to Use Closest to Me Home & Garden Sporting Goods Toys Business & Finance 14:33 Done

THIS IS MORE THAN SUMMER SHOPPING

THIS IS MAKING MEMORIES TO LAST A LIFETIME

Find what inspires you

ASSISTIVE

Zip Code Get a Quote

LET EBAY VAULT SELL

Demonstration

Summary

Always have web accessibility at the back of your mind when building a website. That is ensure you are **building for everyone** that have access to the web.

- [The Teiresias Centre](#) (MUNI)
- [BlindFriendly.cz](#) (TyfloCentrum Brno)
- [Poslepu.cz](#) (Radek Pavlíček)