

Web Accessibility: responsibilities by project role

PROJECT NAME:

PROJECT MANAGER:

VISUAL DESIGNER:

UX / INTERACTION DESIGNER:

DEVELOPER:

CONTENT EDITOR:

PROJECT MANAGER

- 1 **Include accessibility as a project requirement**
Test: Accessibility included in requirements documentation
- 2 **If procuring software, check vendor's accessibility statement**
Test: Is it WCAG 2.0 AA compliant? Are there known defects?
- 3 **Assign responsibilities to project team members**
Test: Write names next to roles on this sheet
- 4 **Add accessibility tasks to the product backlog**
Test: Tasks been added to project tracking software
- 5 **Monitor progress and report**
Test: Reporting to stakeholders about accessibility work

VISUAL DESIGNER

- 1 **Provide contrast between text and background [1.4.3/1.4.11]**
Test: Use the Color Contrast Analyzer (CCA) to check 4.5:1 ratio
- 2 **Don't use colour alone to distinguish links from body text [1.4.1]**
Test: Links are underlined or clearly distinguished from body text
- 3 **Give an indication of keyboard focus [2.4.7]**
Test: style guide shows focussed state of tabs, buttons and links
- 4 **Keep navigation and presentation consistent [3.2.3 / 3.2.4]**
Test: Navigation items and components are used consistently
- 5 **Allow both portrait and landscape orientation [1.3.4]**
Test: style guide shows both portrait and landscape designs
- 6 **Avoid distractions**
Test: Are users encouraged to maintain momentum?

UX / INTERACTION DESIGNER

- 1 **Allow users to pause, stop, or mute audio [1.4.2]**
Test: Are there buttons for controlling audio?
- 2 **Let users extend time limits [2.2.1]**
Test: Do users receive a warning message before timeout?
- 3 **Allow pausing of animations [2.2.2]**
Test: Do animations have a pause button or stop automatically?
- 4 **Avoid content that flashes [2.3.1]**
Test: Is there visual content that changes 4 times per second?
- 5 **Group related content: banner, main, footer [2.4.1]**
Test: Is navigation clearly separated from content?
- 6 **Content displayed on hover can be controlled [3.2.1 / 1.4.13]**
Test: Tab through the page. Do pop-ups obscure your view?
- 7 **Avoid surprise changes when inputting information [3.2.2]**
Test: Are submit buttons provided for input fields?
- 8 **Allow text resizing and spacing [1.4.4 / 1.4.10 / 1.4.12]**
Test: Press CTRL + Plus (+) three times. Press CTRL + 0 to reset.
- 9 **Make navigation order of links and forms logical [2.4.3]**
Test: Check order by tabbing through page from top to bottom
- 10 **Provide an index, sitemap or site search [2.4.5]**
Test: Is there a link to an index, sitemap or search?
- 11 **Make each page navigable by keyboard alone [2.1.1 / 2.5.7]**
Test: Can you complete all tasks without using a mouse?
- 12 **Don't trap the keyboard [2.1.2]**
Test: Can you tab through the entire page?
- 13 **Mobile: Provide alternatives to multipoint gestures [2.5.1]**
Test: Can you operate the device using one finger?

DEVELOPER

- 1 **Use heading tags, table headings and lists [1.3.1]**
Test: Use the a11y-outline bookmarklet for Chrome/Firefox
- 2 **Present content in a meaningful sequence [1.3.2]**
Test: Does the DOM order match the visual order?
- 3 **Identify input formats for form fields [3.3.1]**
Example: Date of birth (dd/mm/yyyy)
- 4 **Clearly identify input roles and errors [1.3.4 / 3.3.3]**
Test: Do form inputs have ARIA roles and error messages?
- 5 **Use labels, fieldsets and legends in forms [3.3.2 / 2.5.4]**
Test: Use the Siteimprove plugin for Chrome
- 6 **Allow users to confirm / reverse changes to data [3.3.4]**
Test: Are users asked to confirm changes to important data?
- 7 **Identify the language of the page [3.1.1 / 3.1.2]**
Test: Does the <html> tag include a lang attribute?
- 8 **Use valid HTML and avoid duplicate IDs [4.1.1]**
Test: Use axe-core accessibility engine to automate testing
- 9 **Provide names and roles for all components [4.1.2]**
Test: Use the Siteimprove plugin for Chrome
- 10 **Keyboard shortcuts can be turned off [2.5.3]**
Test: Can users disable character key shortcuts?
- 11 **Activate functions on mouseup, not mousedown [2.5.2]**
Test: Does release outside a target cancel the click action?
- 12 **Inform assistive technologies of changes to content [4.1.3]**
Test: Use NVDA with Focus Highlight plugin to track focus

Common accessibility problems

© Copyright: The University of Melbourne. 18 May 2018.
Version 2.2. <http://go.unimelb.edu.au/cjv6>

Licensed under the Creative Commons Attribution 3.0 Australia License.



CONTENT EDITOR

- 1 **Add alt text to images [1.1.1]**
Test: What does the image convey? Is it decorative?
- 2 **Don't use images of text [1.4.5]**
Test: Can you select all text with a mouse?
- 3 **Add labels to form fields [2.4.6]**
Test: Does each input have a label above or beside it?
- 4 **Add transcripts to audio [1.2.1]**
Test: How will a deaf person access the audio content?
- 5 **Add captions to video [1.2.2 / 1.2.4]**
Test: Will a deaf / non-native speaker understand?
- 6 **Describe silent content in videos [1.2.3 / 1.2.5]**
Test: How will blind users know what is happening?
- 7 **Don't make instructions reliant on vision [1.3.3]**
Example: Don't say 'Click on top right of screen'
- 8 **Add a meaningful page title [2.4.2]**
Test: Is each page title concise and unique?
- 9 **Avoid 'click here' link text [2.4.4]**
Test: Does link text match the title of the target page?
- 10 **Add informative headings [2.4.6]**
Test: Do the headings alone describe the page content?
- 11 **Avoid jargon [3.1.3]**
Test: Does the web site use plain English?

INSTRUCTIONS

1. Project Manager assigns responsibilities to project team
2. Team members check each item
Tip: numbers in brackets refer to WCAG 2.1 Success Criteria
<https://www.w3.org/WAI/WCAG21/quickref>
3. Team members fix problems and report on progress
4. Project Manager reports to stakeholders