Accessibility in Design Systems: Easy Wins and Lessons Learnt

MEWS

Make it remarkable.

Mews.com

+

Table of Content:

Intro

Riccardo Erra
Senior Frontend Engineer

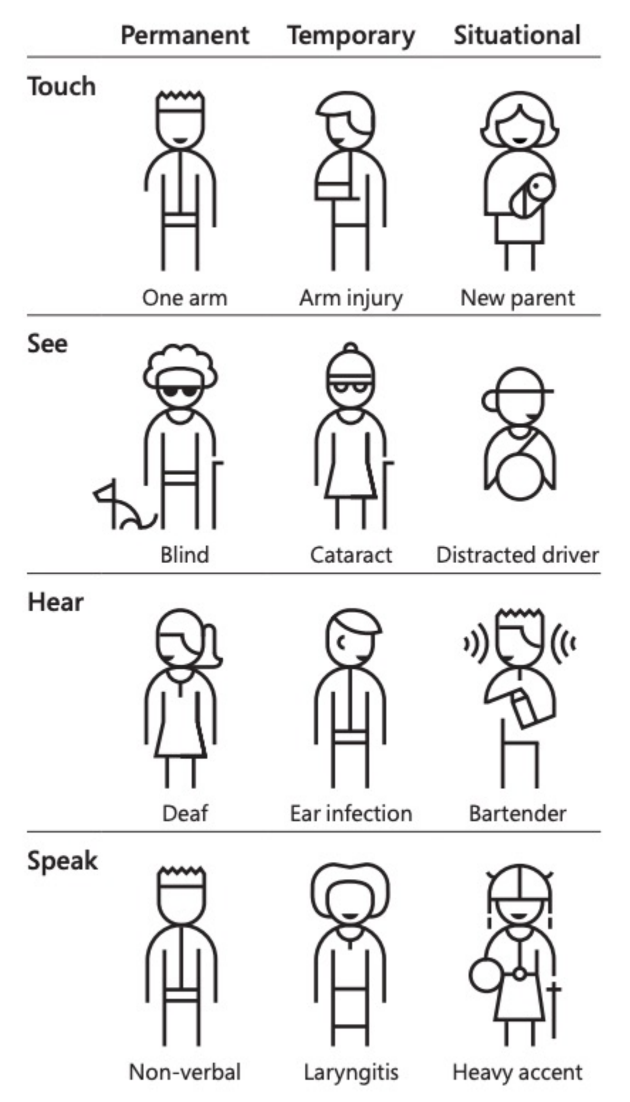
https://riccardoerra.com

Management system for hotels and properties as well as their guests

A11Y-101

Accessibility is the practice of making your websites usable by as many people as possible

Over 1 billion people currently experience some form of disability



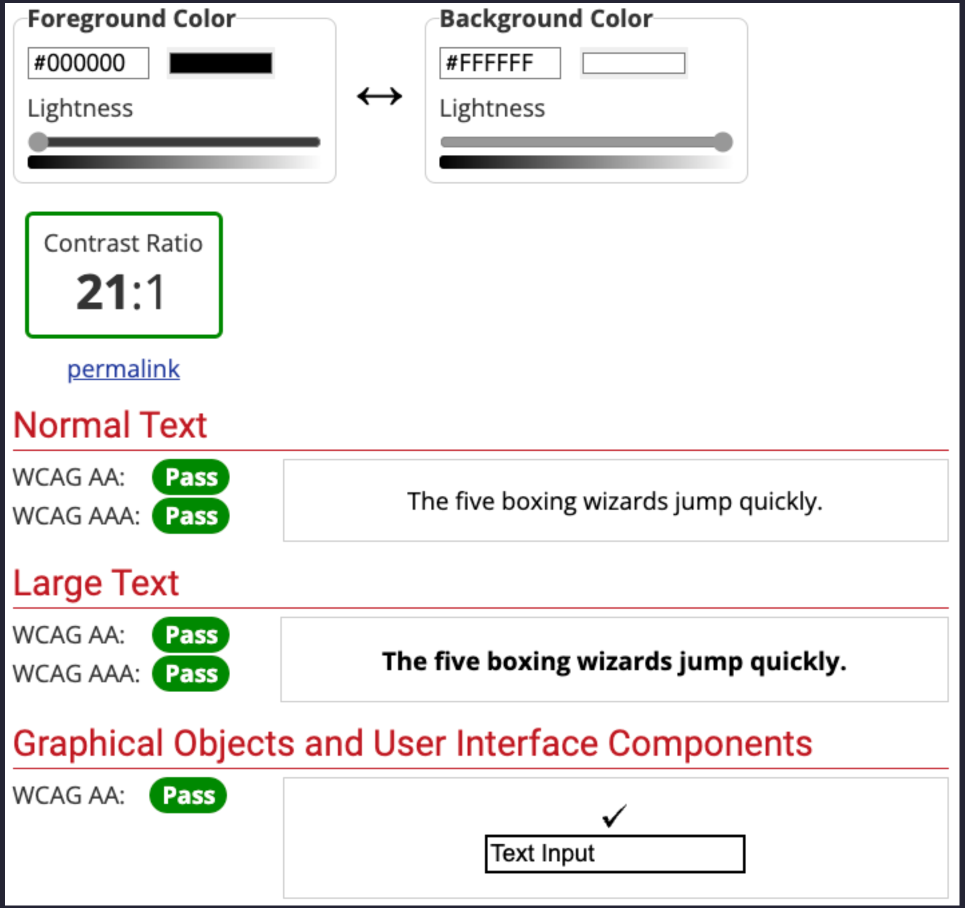
	Permanent	Temporary	Situational
Touch	One arm	Arm injury	New parent
See	Blind	Cataract	Distracted driver
Hear	Deaf	Ear infection	Bartender
Speak	Non-verbal	Laryngitis	Heavy accent

Businesses lose approximately £2B/m by ignoring the needs of disabled people

Easy Wins

Design systems present a unique opportunity to build accessibility and inclusion into your component libraries from the get go

Start with your colours



Foreground Color

#000000

Lightness

↔

Background Color

#FFFFFF

Lightness

Contrast Ratio

21:1

permalink

Normal Text

WCAG AA: **Pass**

WCAG AAA: **Pass**

The five boxing wizards jump quickly.

Large Text

WCAG AA: **Pass**

WCAG AAA: **Pass**

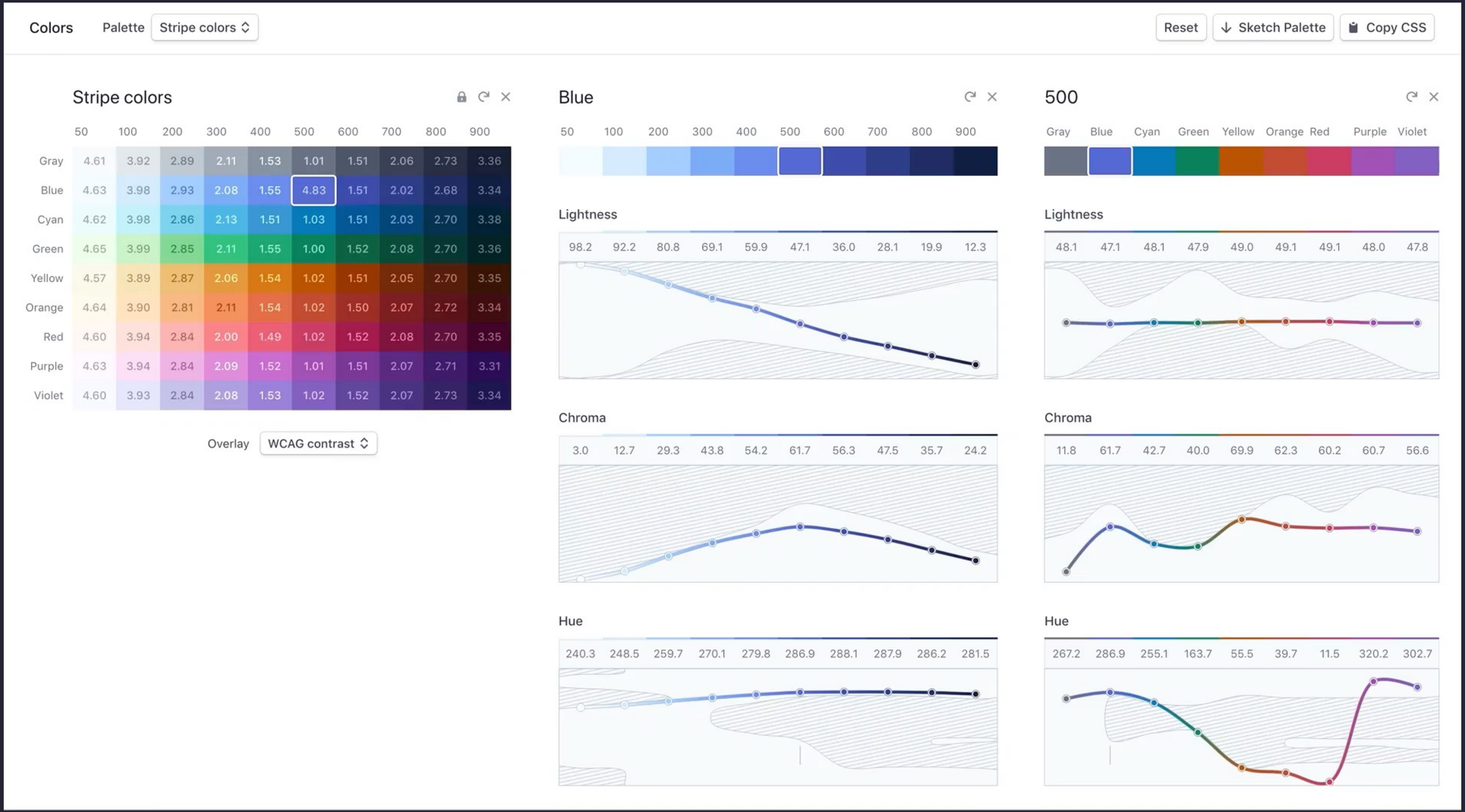
The five boxing wizards jump quickly.

Graphical Objects and User Interface Components

WCAG AA: **Pass**

✓

Text Input



Stripe colors

	50	100	200	300	400	500	600	700	800	900
Gray	4.61	3.92	2.89	2.11	1.53	1.01	1.51	2.06	2.73	3.36
Blue	4.63	3.98	2.93	2.08	1.55	4.83	1.51	2.02	2.68	3.34
Cyan	4.62	3.98	2.86	2.13	1.51	1.03	1.51	2.03	2.70	3.38
Green	4.65	3.99	2.85	2.11	1.55	1.00	1.52	2.08	2.70	3.36
Yellow	4.57	3.89	2.87	2.06	1.54	1.02	1.51	2.05	2.70	3.35
Orange	4.64	3.90	2.81	2.11	1.54	1.02	1.50	2.07	2.72	3.34
Red	4.60	3.94	2.84	2.00	1.49	1.02	1.52	2.08	2.70	3.35
Purple	4.63	3.94	2.84	2.09	1.52	1.01	1.51	2.07	2.71	3.31
Violet	4.60	3.93	2.84	2.08	1.53	1.02	1.52	2.07	2.73	3.34

Overlay: WCAG contrast

Blue

	50	100	200	300	400	500	600	700	800	900
Lightness	98.2	92.2	80.8	69.1	59.9	47.1	36.0	28.1	19.9	12.3
Chroma	3.0	12.7	29.3	43.8	54.2	61.7	56.3	47.5	35.7	24.2
Hue	240.3	248.5	259.7	270.1	279.8	286.9	288.1	287.9	286.2	281.5

500

	Gray	Blue	Cyan	Green	Yellow	Orange	Red	Purple	Violet
Lightness	48.1	47.1	48.1	47.9	49.0	49.1	49.1	48.0	47.8
Chroma	11.8	61.7	42.7	40.0	69.9	62.3	60.2	60.7	56.6
Hue	267.2	286.9	255.1	163.7	55.5	39.7	11.5	320.2	302.7

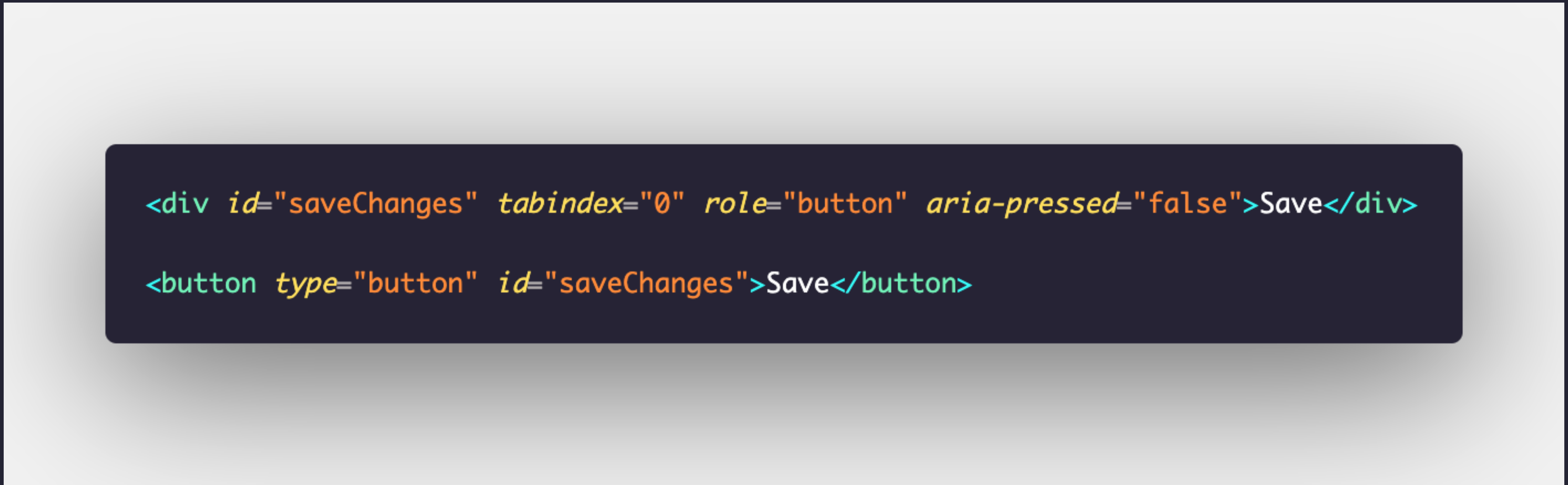
MEWS

Make it remarkable.

Mews.com

+

Web content can be made accessible just by making sure the correct Hypertext Markup Language elements are used



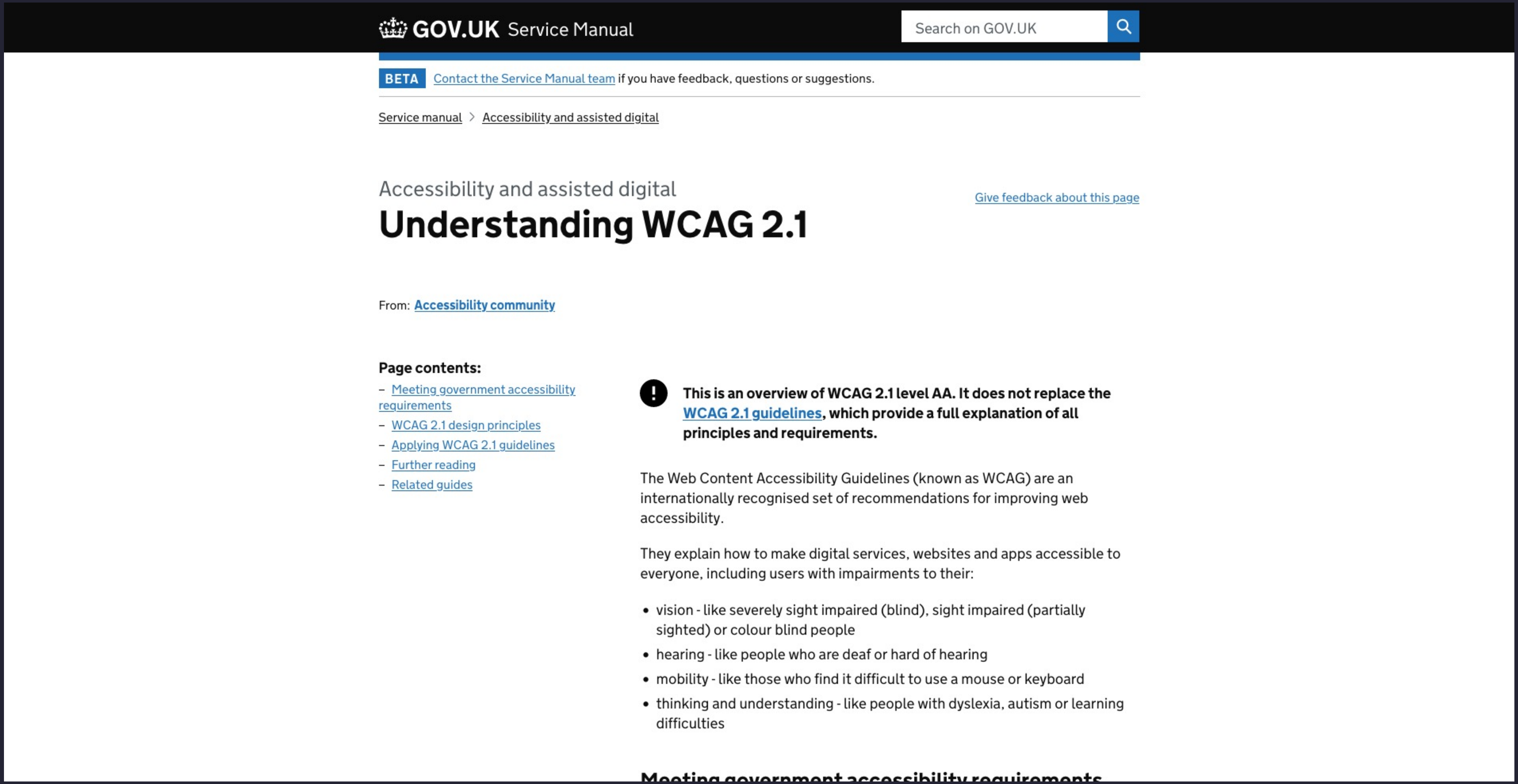
```
<div id="saveChanges" tabindex="0" role="button" aria-pressed="false">Save</div>

<button type="button" id="saveChanges">Save</button>
```

Ensure that all content can be accessed with the keyboard alone

Screen readers enable the independent use of digital devices by those who otherwise would not be able

Lessons Learnt

Learn about standards, guidelines and requirements



BETA Contact the Service Manual team if you have feedback, questions or suggestions.

Service manual > Accessibility and assisted digital

Accessibility and assisted digital

Understanding WCAG 2.1

Give feedback about this page

From: **Accessibility community**

Page contents:

! This is an overview of WCAG 2.1 level AA. It does not replace the WCAG 2.1 guidelines, which provide a full explanation of all principles and requirements.

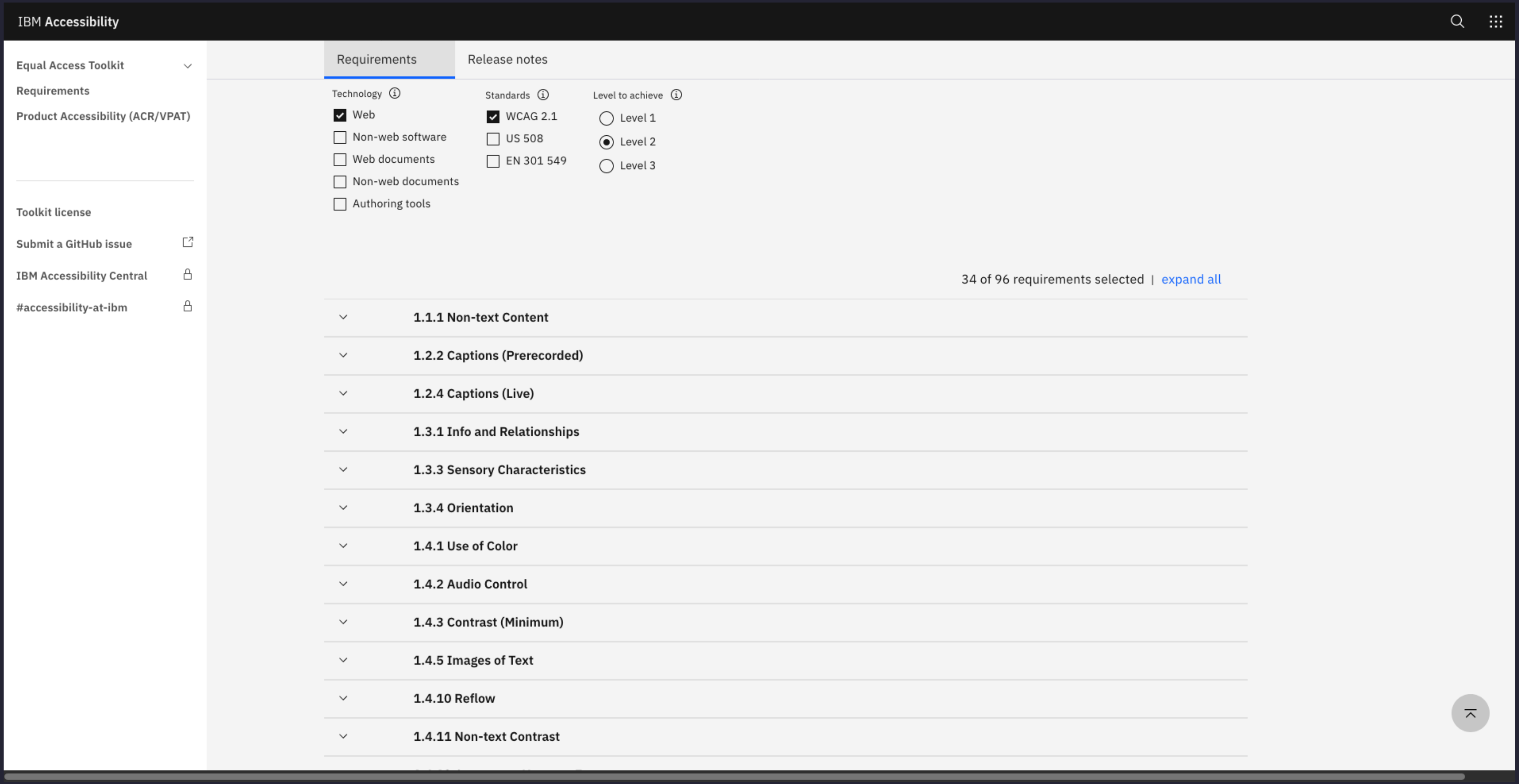
The Web Content Accessibility Guidelines (known as WCAG) are an internationally recognised set of recommendations for improving web accessibility.

They explain how to make digital services, websites and apps accessible to everyone, including users with impairments to their:

- vision - like severely sight impaired (blind), sight impaired (partially sighted) or colour blind people
- hearing - like people who are deaf or hard of hearing
- mobility - like those who find it difficult to use a mouse or keyboard
- thinking and understanding - like people with dyslexia, autism or learning difficulties

Meeting government accessibility requirements

+



Technology

☑ Web
□ Non-web software
□ Web documents
□ Non-web documents
□ Authoring tools

Standards

☑ WCAG 2.1
□ US 508
□ EN 301 549

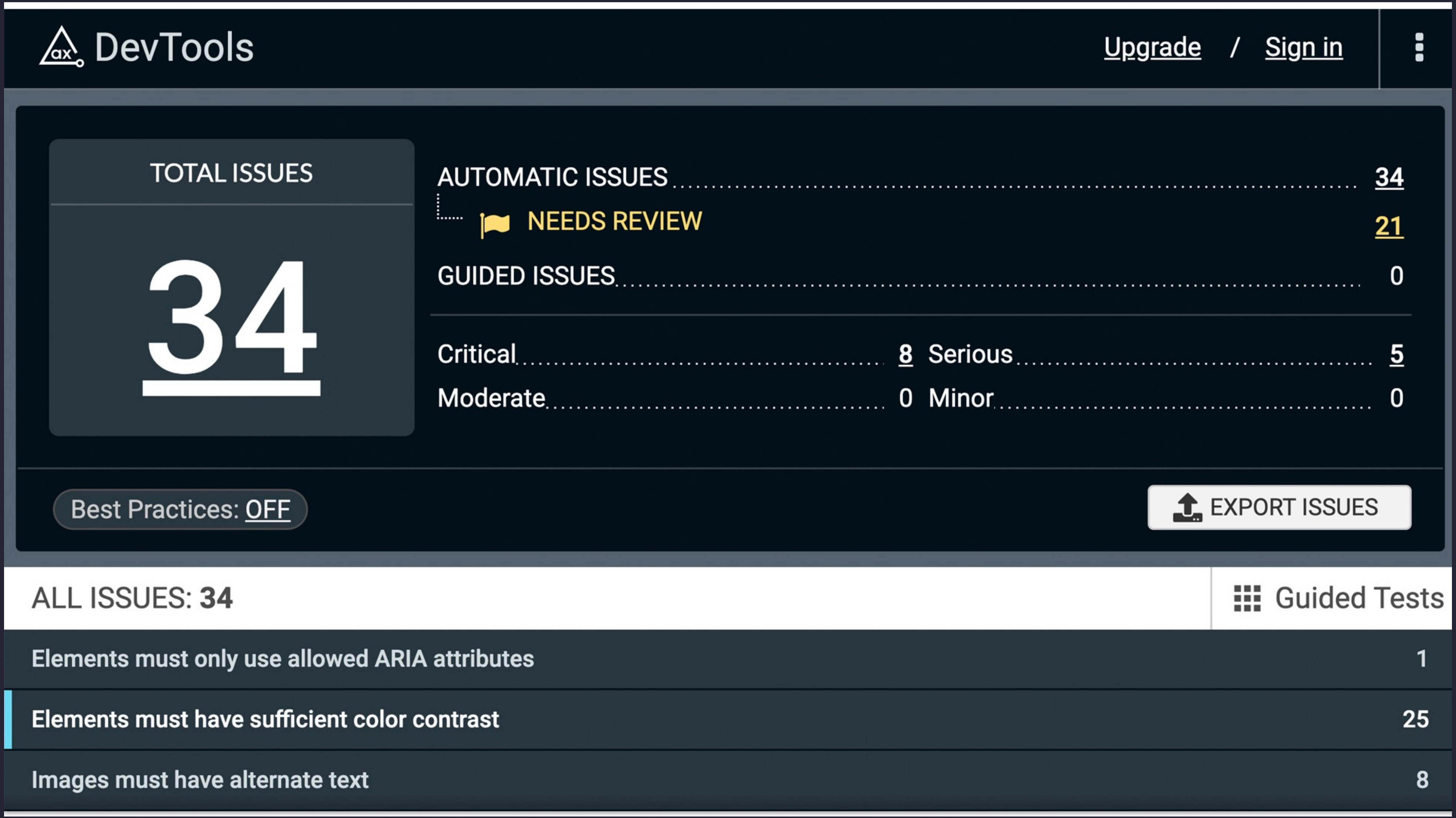
Level to achieve

○ Level 1
● Level 2
○ Level 3

34 of 96 requirements selected | expand all

- 1.1.1 Non-text Content
- 1.2.2 Captions (Prerecorded)
- 1.2.4 Captions (Live)
- 1.3.1 Info and Relationships
- 1.3.3 Sensory Characteristics
- 1.3.4 Orientation
- 1.4.1 Use of Color
- 1.4.2 Audio Control
- 1.4.3 Contrast (Minimum)
- 1.4.5 Images of Text
- 1.4.10 Reflow
- 1.4.11 Non-text Contrast

Never stop improving your testing skills



TOTAL ISSUES

34

AUTOMATIC ISSUES 34

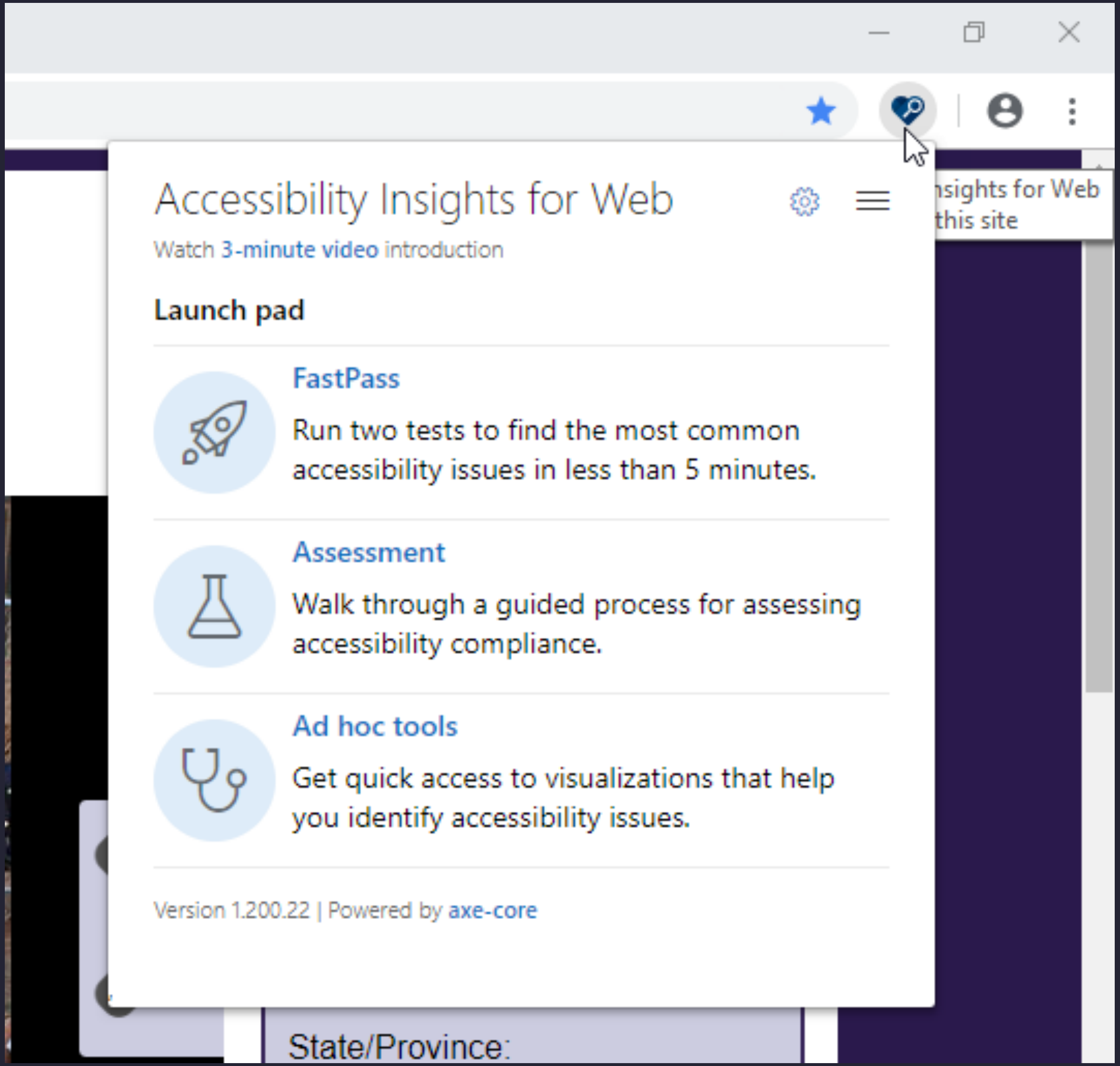
NEEDS REVIEW 21

GUIDED ISSUES 0

Critical	8	Serious	5
Moderate	0	Minor	0

ALL ISSUES: **34**

Elements must only use allowed ARIA attributes	1
Elements must have sufficient color contrast	25
Images must have alternate text	8



Accessibility Insights for Web

Watch 3-minute video introduction

Launch pad

FastPass

Run two tests to find the most common accessibility issues in less than 5 minutes.

Assessment

Walk through a guided process for assessing accessibility compliance.

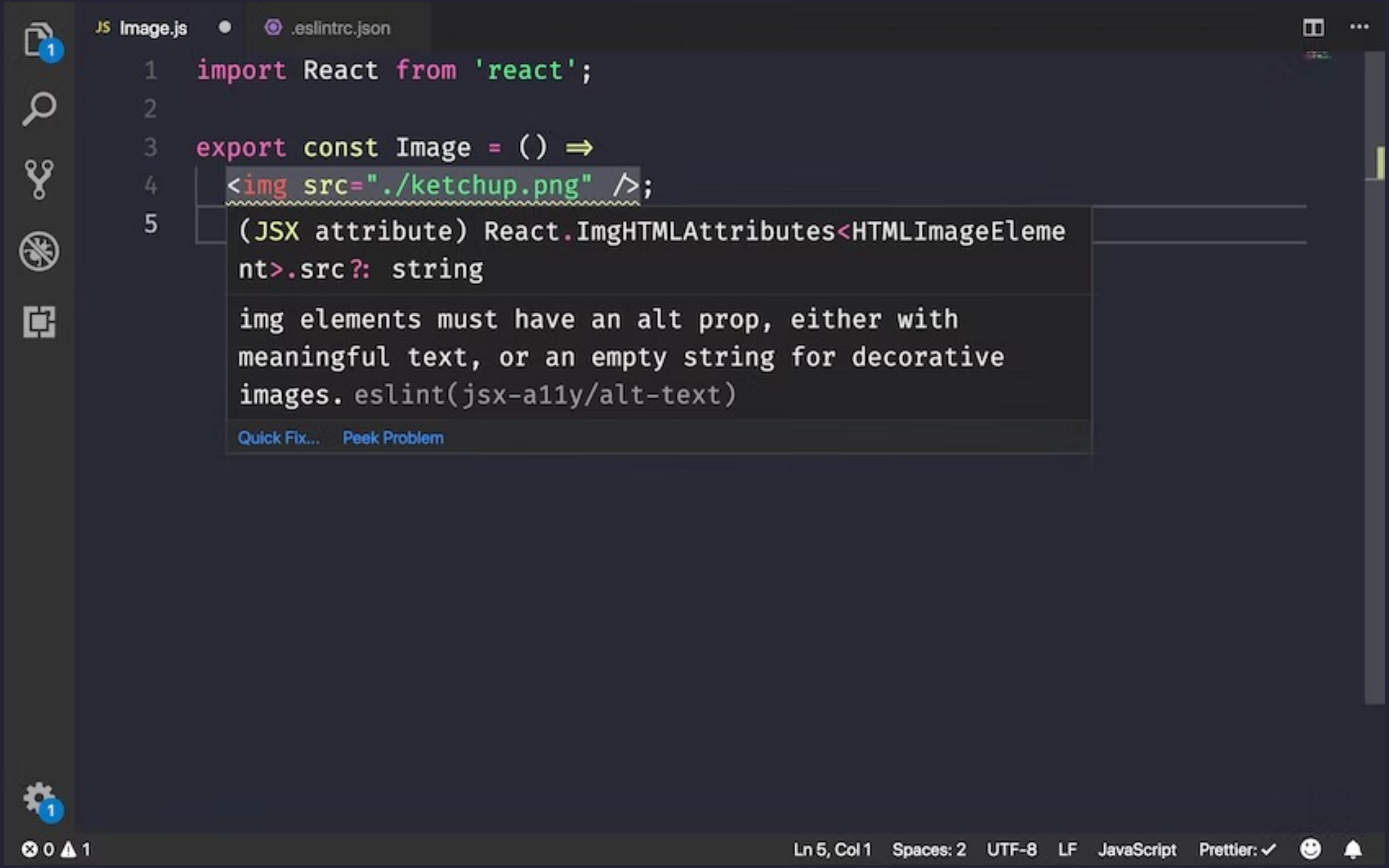
Ad hoc tools

Get quick access to visualizations that help you identify accessibility issues.

Version 1.200.22 | Powered by axe-core

Quick disclaimer

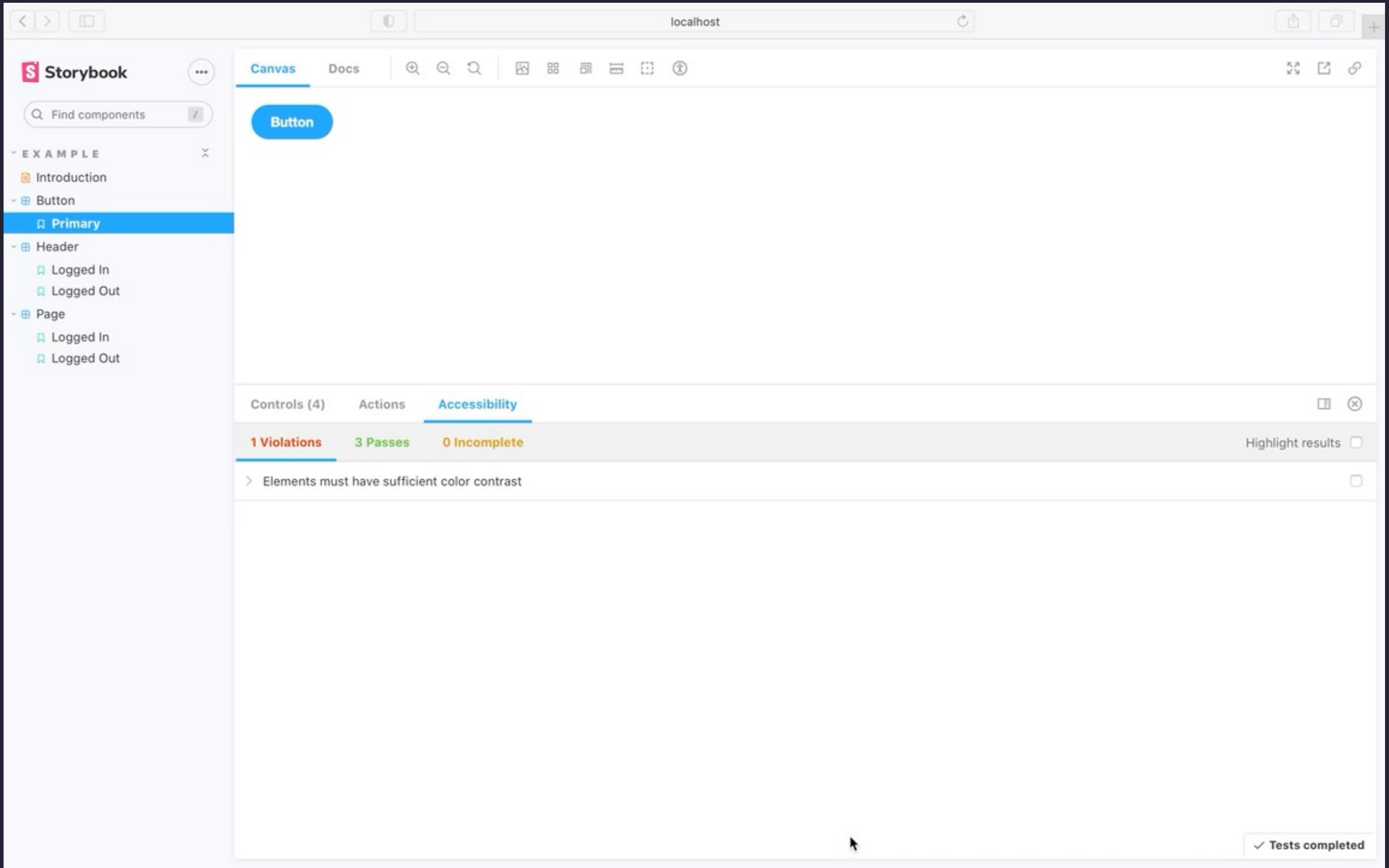
Take advantage of the available tools



```
import React from 'react';

export const Image = () ⇒
  <img src="./ketchup.png" />;

```

(JSX attribute) React.ImgHTMLAttributes<HTMLImageElement>.src?: string

img elements must have an alt prop, either with meaningful text, or an empty string for decorative images. eslint(jsx-a11y/alt-text)

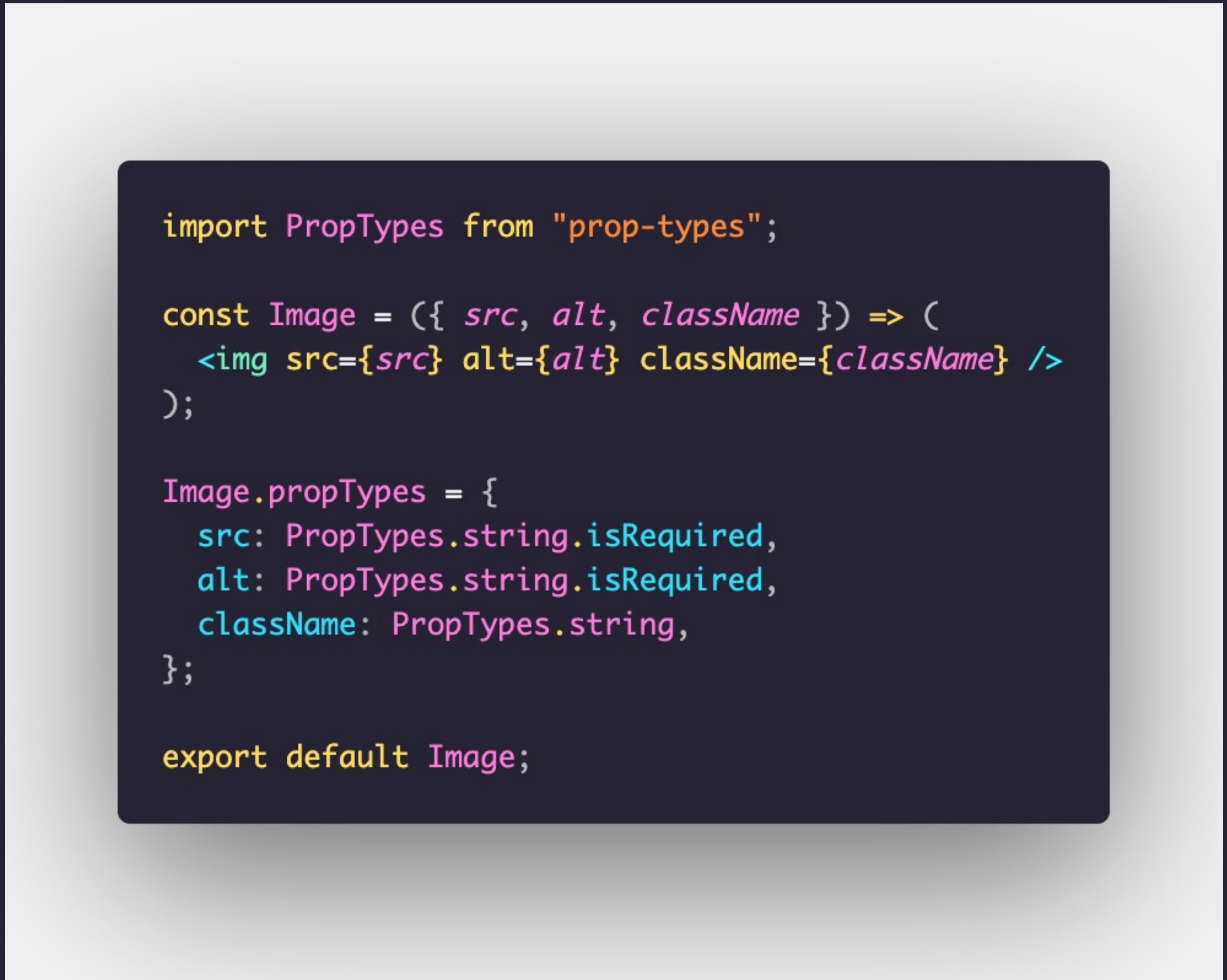


1 Violations 3 Passes 0 Incomplete

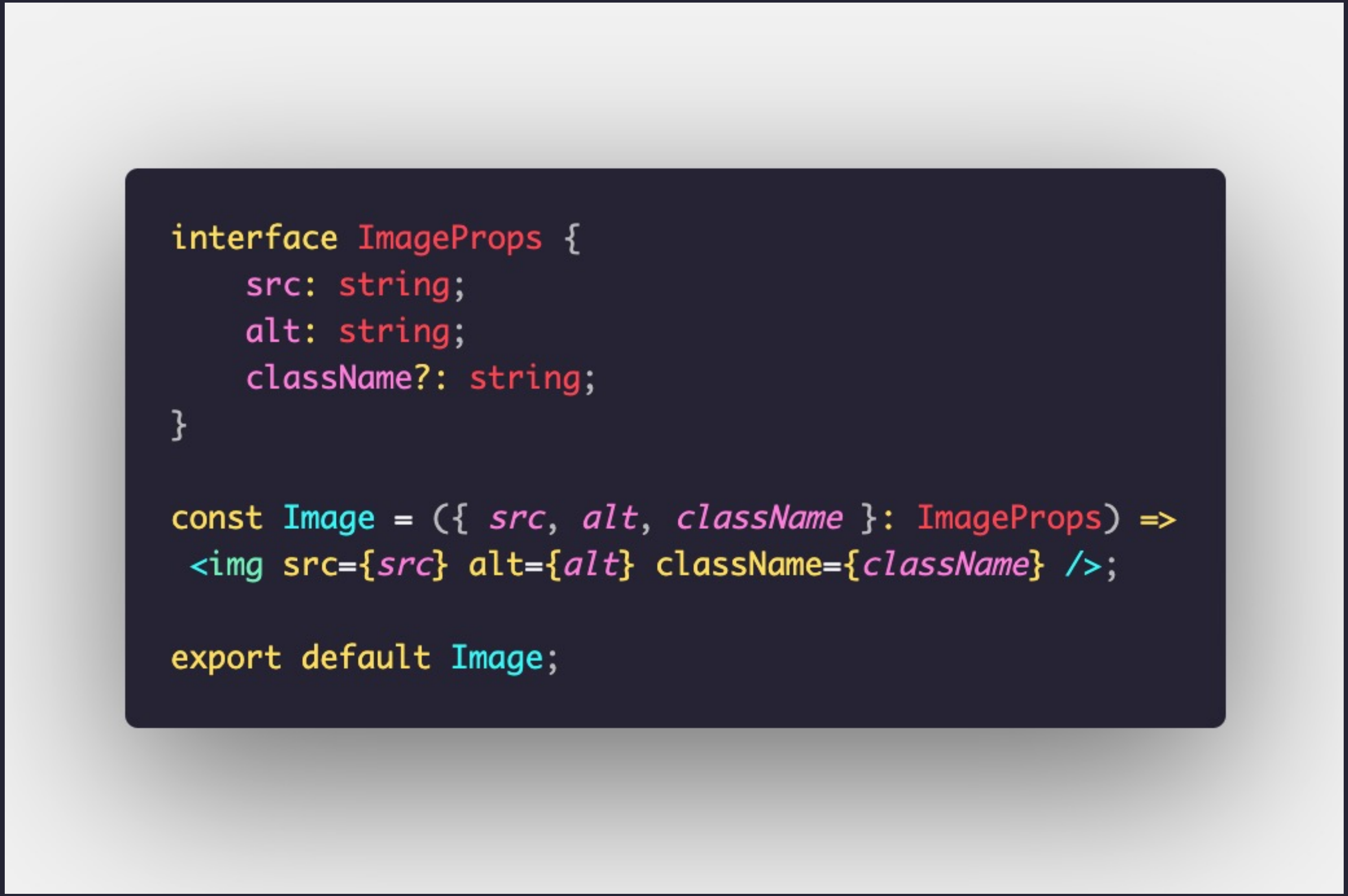
Elements must have sufficient color contrast

✓ Tests completed

Use the force aka provide the right data or you'll get yelled at by your computer



```
import PropTypes from "prop-types";

const Image = ({ src, alt, className }) => (
  <img src={src} alt={alt} className={className} />
);

Image.propTypes = {
  src: PropTypes.string.isRequired,
  alt: PropTypes.string.isRequired,
  className: PropTypes.string,
};

export default Image;
```



```
interface ImageProps {
    src: string;
    alt: string;
    className?: string;
}

const Image = ({ src, alt, className }: ImageProps) =>
 <img src={src} alt={alt} className={className} />;

export default Image;
```

Accessible component ≠ Accessible page

Don’t forget to commission a proper professional audit

Outro

The work is never done but a bit of effort is better than no effort at all

Your most powerful weapon is empathy

Thank you!

We are hiring!
We are looking for a Senior Product Manager in the Design System team

Resources

- Slides: https://github.com/riccardoerra/accessibility-design-systems/blob/main/accessibility-in-design-systems.pdf
- Accessibility definition: https://developer.mozilla.org/en-US/docs/Web/Accessibility
- Disabilities illustration: https://devblogs.microsoft.com/xamarin/the-journey-to-accessible-apps
- Financial statistic: https://wearepurple.org.uk/the-purple-pound-infographic
- Design systems and accessibility: https://xd.adobe.com/ideas/principles/design-systems/accessibility-in-design-systems
- Contrast checker: https://webaim.org/resources/contrastchecker
- Stripe colour system: https://stripe.com/blog/accessible-color-systems
- HTML and accessibility: https://developer.mozilla.org/en-US/docs/Learn/Accessibility/HTML
- ARIA button example: https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles/button_role
- Keyboard accessibility: https://webaim.org/techniques/keyboard
- Screen readers accessibility: https://blog.hubspot.com/website/screen-reader-accessibility

- GOV.UK understanding WCAG: https://www.gov.uk/service-manual/helping-people-to-use-your-service/understanding-wcag
- IBM accessibility requirements: https://www.ibm.com/able/requirements/requirements
- GOV.UK accessibility statement: https://design-system.service.gov.uk/accessibility
- Deque axe DevTools extension: https://www.deque.com/axe/devtools/chrome-browser-extension
- Accessibility Insights: https://accessibilityinsights.io/
- Accessibility auditing with React: https://web.dev/accessibility-auditing-react
- Accessibility tests with Storybook: https://storybook.js.org/docs/react/writing-tests/accessibility-testing
- Enforcing accessibility best practices: https://bradfrost.com/blog/post/enforcing-accessibility-best-practices-with-component-proptypes
- Senior Product Manager role: https://grnh.se/914aa322teu