

Testing Web Accessibility

How to find accessibility vulnerabilities in your website

Adrián Bolonio

 **@bolonio**

Adrián Bolonio



Madrid (Spain)



**Web Team Lead &
Frontend Developer**



#a11y Advocate



@bolonio



Agenda

- **Accessibility (a11y)**
- **Web Accessibility**
- **Developing Web Accessibility**
- **Testing Web Accessibility**
 - **Testing the code**
 - **Testing the DOM**
 - **Testing in the browser**

Accessibility (a11y)

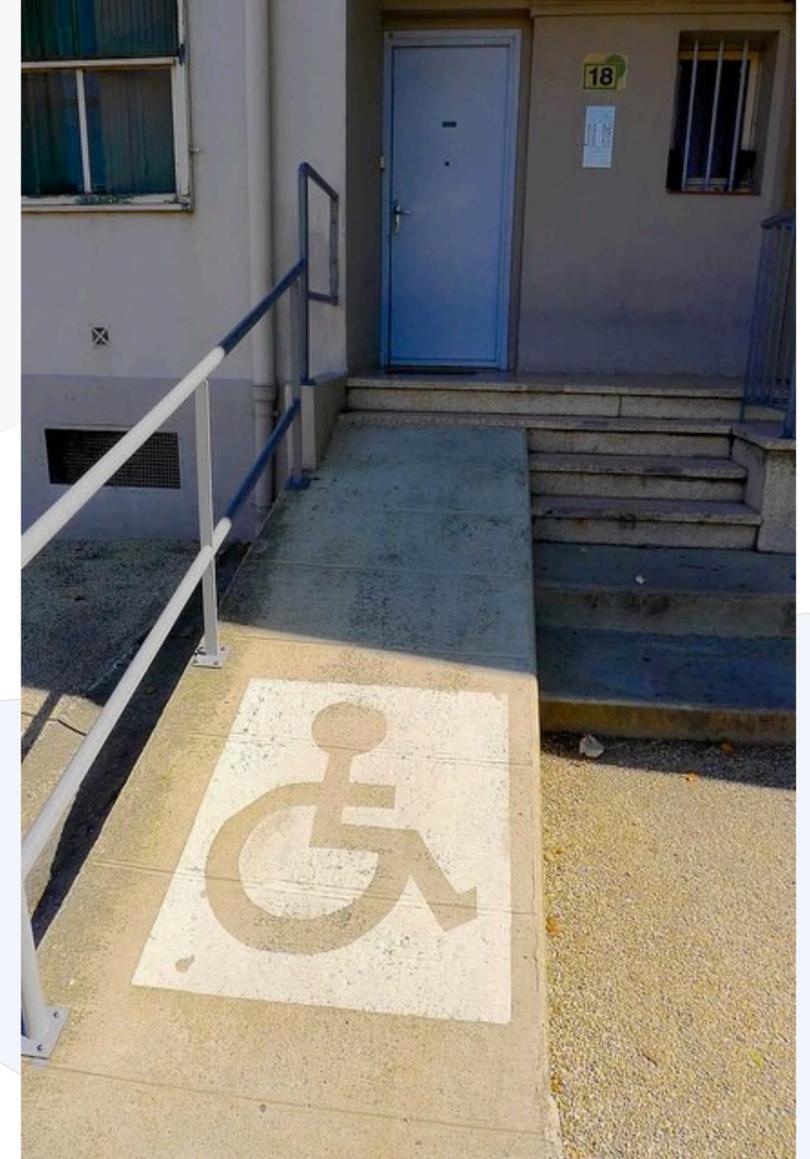
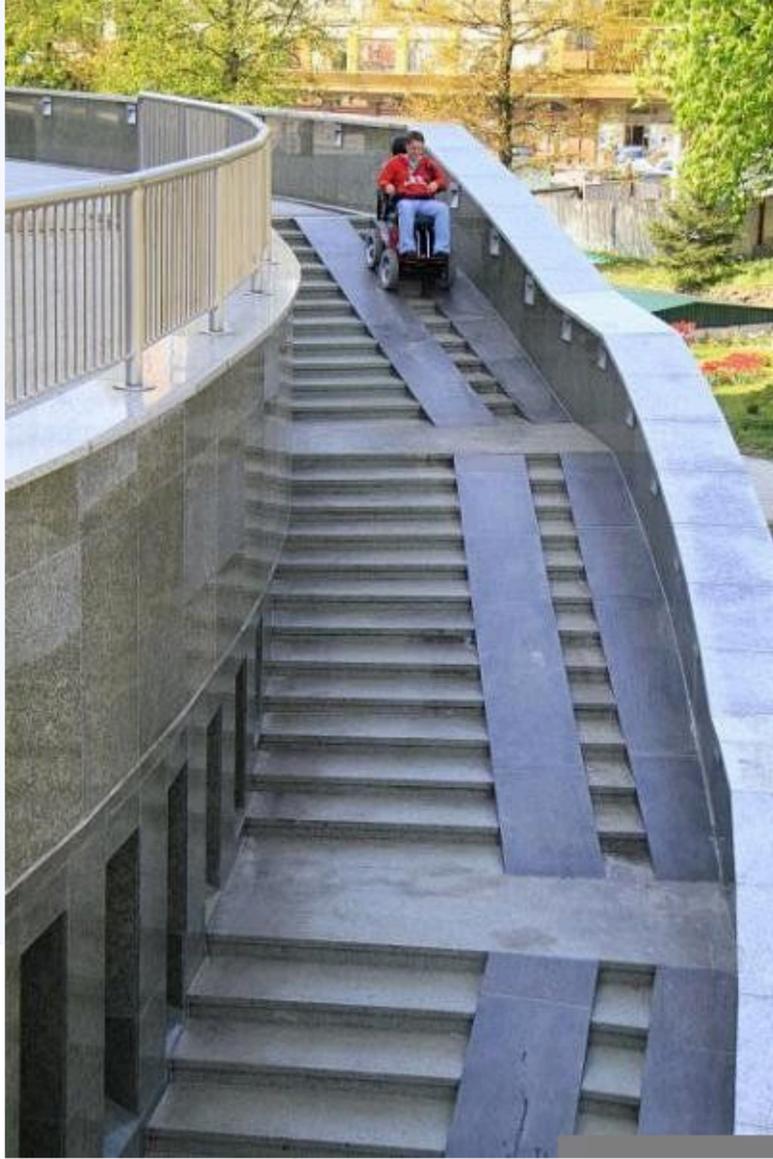
Accessibility (a11y)

accessibility

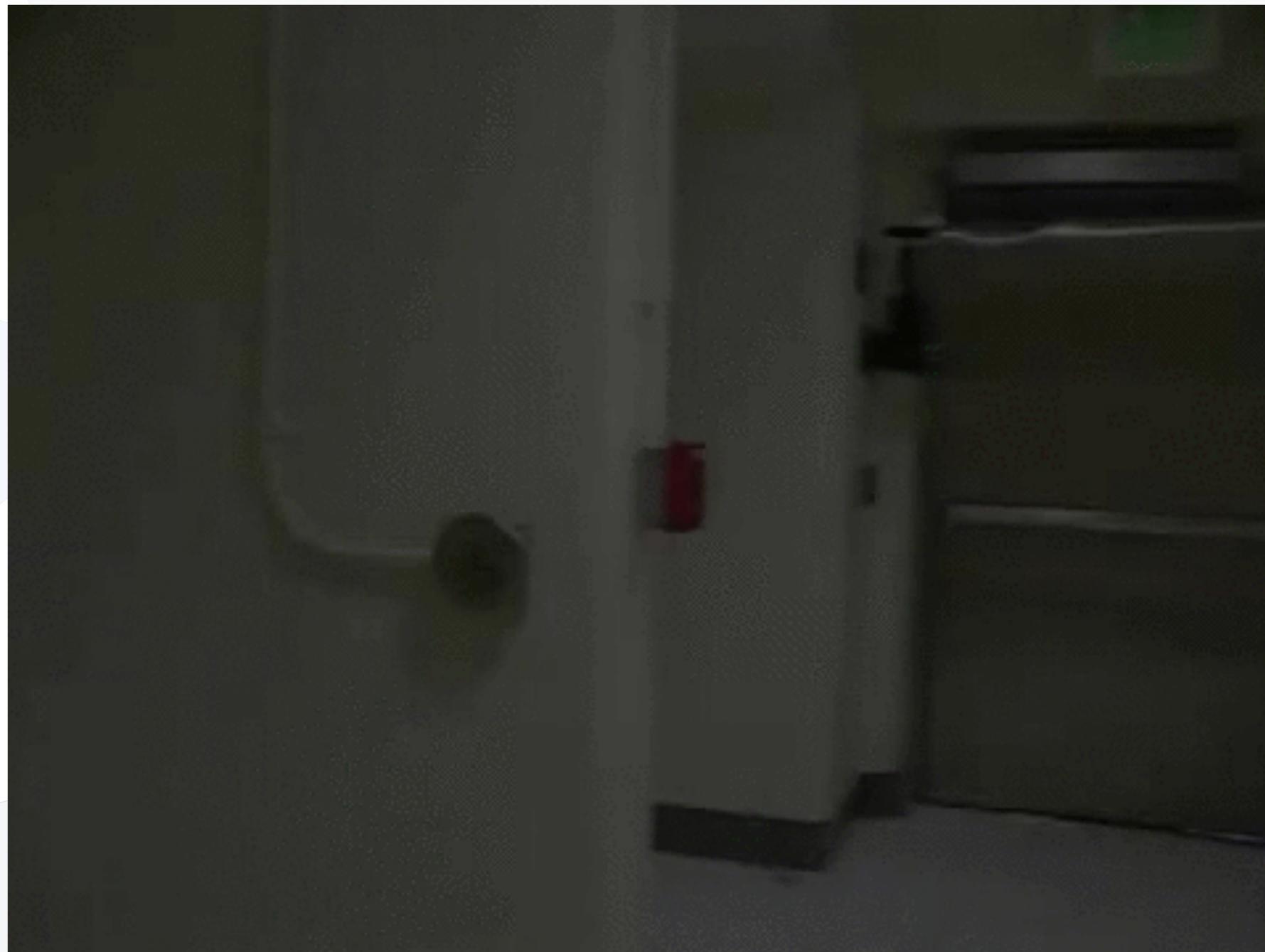
1 2 3 4 5 6 7 8 9 10 11

a11y

Accessibility (a11y)



Accessibility (a11y)



Web Accessibility (a11y)

Web Accessibility (a11y)

The Web is fundamentally designed to work for all people, whatever their hardware, software, language, location, or ability. When the Web meets this goal, it is **accessible** to people with a diverse range of hearing, movement, sight, and cognitive ability.

Accessibility is essential for developers and organizations that want to create high-quality websites and web tools, and **not exclude** people from using their products and services.

Web accessibility means that websites, tools, and technologies are designed and developed so that **people with disabilities can use them.**

“The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect.”

Tim Berners-Lee, W3C Director and inventor of the World Wide Web

Web Accessibility (a11y)



You need to click on the button in the **top right** corner.

You need to click on the button with the **engine icon**.

Web Accessibility (a11y)

2018 7,6 billion

1+ billion

people live with some form of vision impairment

Developing Web Accessibility

How does you website “sound” like?

Screen Readers

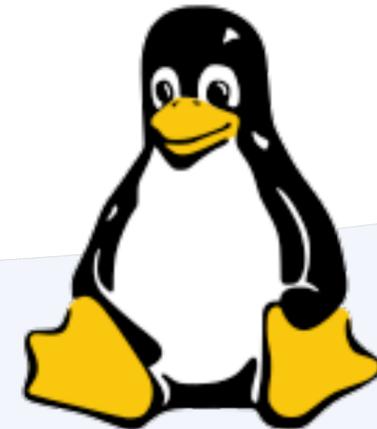
Screen readers are software programs that allow blind or visually impaired users to read the text that is displayed on the computer screen with a speech synthesizer.



Voice Over



Narrator



Orca

Developing Web Accessibility

Best practices

Language

Defining the correct **natural language** in an HTML page is very important, because it helps assistive technology (screen readers) to choose the correct voice profile or character set.

Always use a language attribute on the html element. This is inherited by all other elements, and so will set a default language for the text in the document head element.

Note that you should use the html element rather than the body element, since the body element doesn't cover the text inside the document's head element.



```
<html lang="en">
  <body>
    Select your link:
    <a href="/en">Use this link to go to a new page</a>
    <a href="/ru">Используйте эту ссылку, чтобы перейти на новую страницу</a>
  </body>
</html>
```

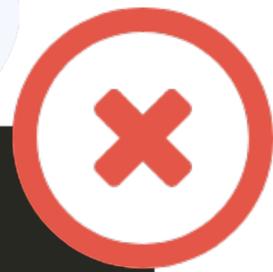


```
<html lang="en">
  <body>
    Select your link:
    <a href="/en">Use this link to go to a new page</a>
    <a href="/ru" lang="ru">Используйте эту ссылку, чтобы перейти на новую страницу</a>
  </body>
</html>
```

Image alt

Every image **must** have an alt attribute. This is a requirement of HTML standard (with perhaps a few exceptions in HTML5).

Images without an alt attribute are likely **inaccessible**. In some cases, images may be given an empty or null alt attribute (e.g., alt="").



```
<html lang="en">
  <body>
    
  </body>
</html>
```



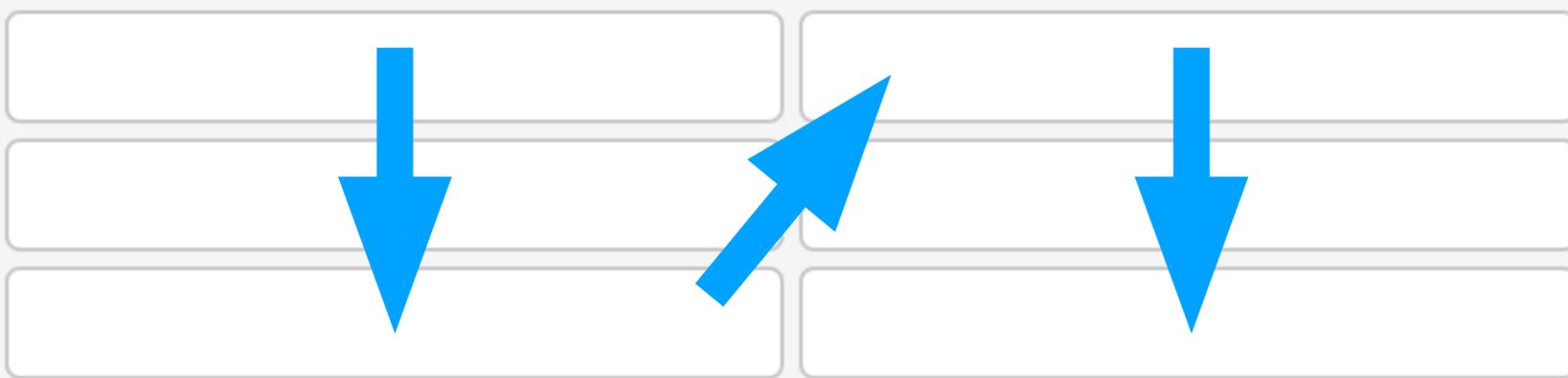
```
<html lang="en">
  <body>
    
  </body>
</html>
```

Tab Order

Demo

The intent of the **tab order** is to ensure that when users navigate sequentially through content, they encounter information in an **order** that is **consistent** with the meaning of the content and can be operated from the **keyboard (tab key)**. This reduces confusion by letting users form a consistent mental model of the content.

Search for marriage records		
Search criteria	Groom	Bride
First name	<input type="text"/>	<input type="text"/>
Last name	<input type="text"/>	<input type="text"/>
Place of birth	<input type="text"/>	<input type="text"/>



ARIA

Accessible Rich Internet Applications

Aria (Accessible Rich Internet Applications)

Accessible Rich Internet Applications (ARIA) is a set of attributes that define ways to make Web content and Web applications (especially those developed with JavaScript) more accessible to people with disabilities.

ARIA enables developers to describe their widgets in more detail by adding special attributes to the markup.

- ARIA attributes don't affect anything in the webpage.
- Use the correct semantic HTML elements, don't use it as fixer.
- Use aria-* attributes, but test them before using them.

Aria Rule #1

Don't use ARIA, use native HTML instead

If you can use a native HTML element or attribute with the semantics and behaviour you require **already built in**, instead of re-purposing an element and adding an ARIA role, state or property to make it accessible, **then do so**.

If you need a button, use the **<button>** element. Buttons are: Focusable, Clickable (with mouse and keys), and Screen readers identify them as buttons



```
<div role="button">Click Me</div>
```



```
<button>Click Me</button>
```

Aria Rule #2

Do not change native semantics, unless you really have to.

Most of the HTML elements or attributes convey one or other semantics. We are not supposed to change the native semantics unless it is really essential.

An example of this would be if a developer wants to build a heading that is a tab:



```
<h2 role=tab>heading tab</h2>
```



```
<div role=tab><h2>heading tab</h2></div>
```

Aria Rule #3

All interactive ARIA controls must be usable with the keyboard.

If you create a widget that a user can click or tap or drag or drop or slide or scroll, a user must also be able to navigate to the widget and perform an equivalent action using the **keyboard**.

All interactive widgets must be scripted to respond to standard keystrokes or keystroke combinations where applicable.

For example, if creating the **custom** control with **role=button** the element must be able to receive **focus** and a user must be able to activate the action associated with the element using both the **enter** and the **space key**.

Source: <https://www.w3.org/TR/using-aria/>

Aria Rule #4

Do not use `role="presentation"` or `aria-hidden="true"` on a focusable element.

Using either of these on a focusable element will result in some users focusing on 'nothing'.



```
<button role="presentation">  
  Click Me  
</button>  
  
<button aria-hidden="true">  
  Click Me  
</button>
```



```
<button role="presentation" tabindex="-1">  
  Don't Click Me  
</button>  
  
<button aria-hidden="true" style="display: none;">  
  Don't Click Me  
</button>
```

Aria Rule #5

All interactive elements must have an accessible name.

An interactive element only has an accessible name when its Accessibility API accessible name (or equivalent) property has a value.



```
<span>user name</span>  
<input type="text">
```



```
<label for="username">user name</label>  
<input type="text" id="username">
```

Aria Rules

- #1 Don't use ARIA, use native HTML instead**
- #2 Do not change native semantics, unless you really have to.**
- #3 All interactive ARIA controls must be usable with the keyboard.**
- #4 Do not use `role="presentation"` or `aria-hidden="true"` on a focusable element .**
- #5 All interactive elements must have an accessible name.**

Aria-* attributes

Periodic Table of ARIA 1.0 attributes

aria-describedby	aria-required				aria-hidden	aria-busy	aria-valuenow	
aria-label	aria-multiline	aria-live	aria-atomic	aria-has-popup	aria-invalid	aria-disabled	aria-valuemin	
aria-labelledby	aria-multiselectable	aria-relevant	aria-controls	aria-auto-complete	aria-expanded	aria-checked	aria-valuemax	
	aria-orientation	aria-owns	aria-dropeffect	aria-flowto	aria-selected	aria-pressed	aria-setsize	
	aria-sort	aria-readonly			aria-grabbed	aria-active-descendant	aria-level	aria-posinset

Name
 Value

Role

State

Legend

Interactive elements

Colors or Patterns?

Colors Patterns

Aria roles

Periodic Table of ARIA 1.0 Roles

link										main	img
alert	alertdialog	menuitem-radio	tab	treeitem	form	textbox	article	group	presentation	banner	grid
log	dialog	progressbar	tabpanel	menu	button	spinbutton	column-header	heading	region	complementary	gridcell
status	menuitem	scrollbar	timer	menubar	checkbox	combobox	definition	list	separator	contentinfo	row
marquee	menuitem-checkbox	slider	tooltip	tablist	option	radio-group	directory	listitem	toolbar	navigation	row-header
	application	treegrid	tree	radio	listbox	document	math	note	search		

Legend

<input checked="" type="checkbox"/> Forms	<input type="checkbox"/> Tabular data	<input type="checkbox"/> Document sections	<input type="checkbox"/> Interactive elements
<input type="checkbox"/> Embedding content	<input type="checkbox"/> Text-level semantics		

Colors or Patterns?
 Colors Patterns

<... aria-label="" />

The purpose of this technique is to provide a label for objects that can be read by assistive technology. The **aria-label** attribute provides the text label for an object, such as a button.

When a screen reader encounters the object, the aria-label text is read so that the user will know what it is.



```
<html lang="en">
  <body>
    <button onclick="close()">X</button>
  </body>
</html>
```



```
<html lang="en">
  <body>
    <button aria-label="Close" onclick="close()">X</button>
  </body>
</html>
```

<button />

Demo

Aria Rule #1 - Don't use ARIA, use native HTML instead

If you can use a native HTML element or attribute with the semantics and behaviour you require **already built in**, instead of re-purposing an element and adding an ARIA role, state or property to make it accessible, **then do so**.

If you need a button, use the **<button>** element. Buttons are: Focusable, Clickable (with mouse and keys), and Screen readers identify them as buttons



```
<div role="button">Click Me</div>
```

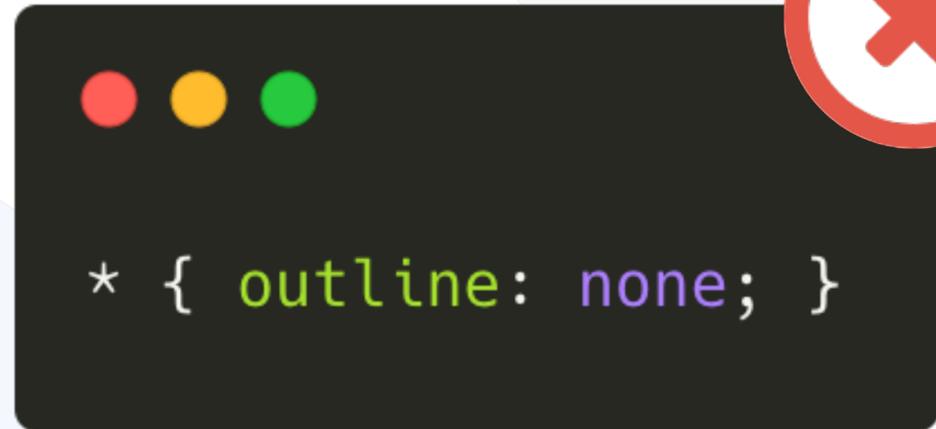


```
<button>Click Me</button>
```



:focus & :focus-visible

Demo



```
* { outline: none; }
```

DO NOT DO IT

Unless you provide custom styles

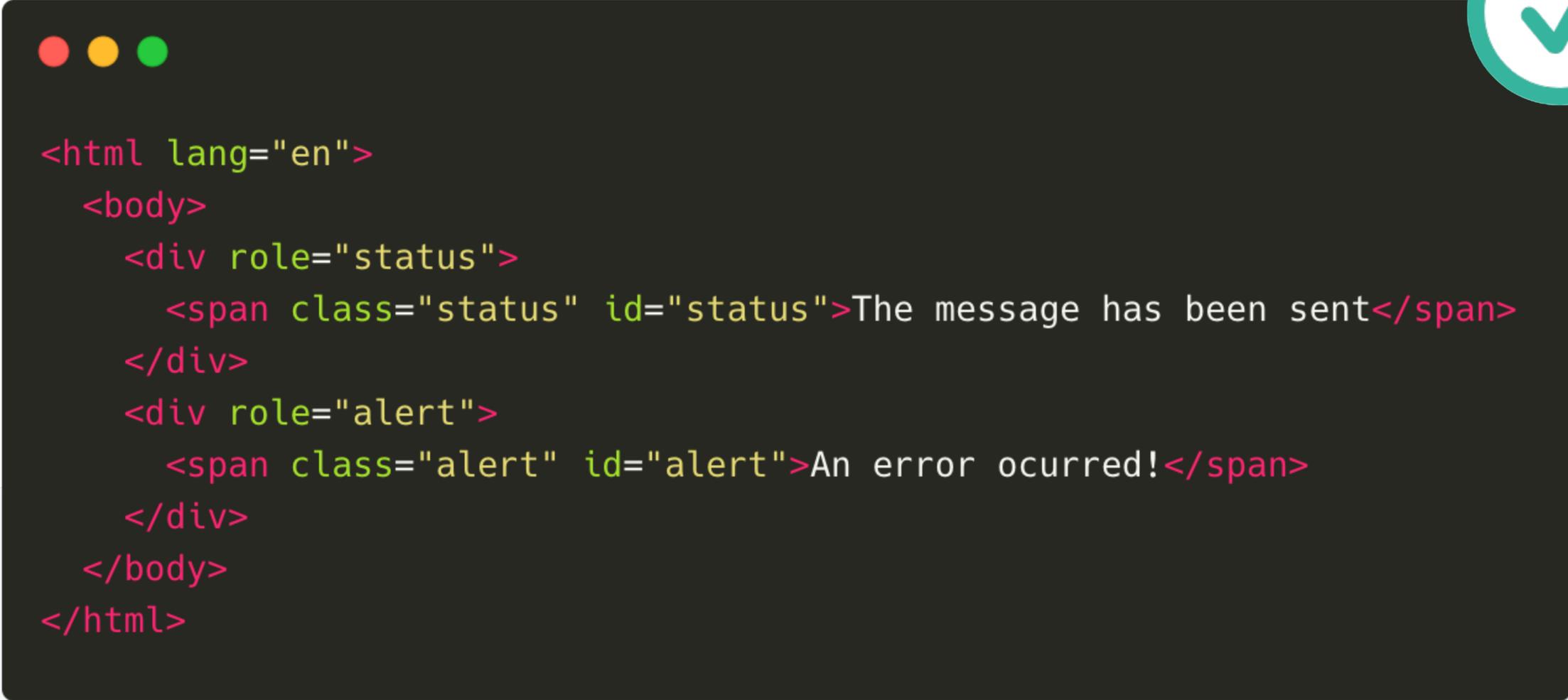
It provides **visual feedback for links that have "focus"** when navigating a web document using the **TAB key** (or equivalent). This is especially useful for users who can't use a mouse or have a visual impairment. If you remove the outline you are making your site inaccessible for these people.

JavaScript

Javascript

Demo

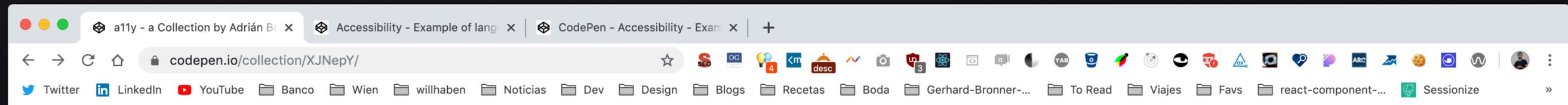
Usually, screen readers only announce content when an element is focused or the user uses the screen reader's own navigation commands. Screen reader users must be **informed** when content changes dynamically, this is done with **ARIA Live Regions**.



```
<html lang="en">
  <body>
    <div role="status">
      <span class="status" id="status">The message has been sent</span>
    </div>
    <div role="alert">
      <span class="alert" id="alert">An error occurred!</span>
    </div>
  </body>
</html>
```

- **Define the correct natural language in an HTML page.**
- **Every `` must have an `alt=` attribute, but it needs to be relevant.**
- **The tab order of your website must be consistent with the meaning of the content.**
- **ARIA - Accessible Rich Internet Applications**
 - **Use the native HTML elements or attributes, try not to use ARIA.**
 - **When using `aria-*` attributes, test it before using them.**
 - **Use a `<button>` element if you need a button.**
- **Do not delete the focus outline, unless you provide custom styles.**
- **Screen reader users must be informed when content changes dynamically, with ARIA Live Regions.**

https://codepen.io/collection/XJNepY/



CODEPEN

CREATE

- Pen
from template
- Project
- Post
- Collection

YOUR

- Dashboard
- Profile
- Pinned Items
- Activity

EXPLORE

- Picks
- Popular
- Following
- Topics

GROW

- Jobs
- CodePen PRO
- Challenges

- Newsletter
- Blog
- Docs & Support

ally

Adrián Bolonio

NOVEMBER 15, 2019

Delete Collection

Edit Collection



Javascript - example of alerts

(aria-live=polite)

Sending a message is important, so I will do it for you, so please click this button

(aria-live=assertive)

Sending a message is important, so I will do it for you, so please click this button

Accessibility - Javascript

Adrián Bolonio

Example of focus

:focus

Click me

:focus-visible

Click me

Via chrome://flags/

Accessibility - Example of fo...

Adrián Bolonio

Example of language detection

English lang="en"

EN: Use this link to go to a new page

Russian lang="ru"

RU: Use this link to go to a new page

RU: Используйте эту ссылку, чтобы перейти на новую страницу

EN: Используйте эту ссылку, чтобы перейти на новую страницу

Language mix

Accessibility - Example of la...

Adrián Bolonio

Example of Buttons

Anchor vs Button

Continue (link) Continue (role=button) Continue

Disabled buttons (not focusable)

Continue Continue

Example of tab order

Custom order (tabindex)

Search for marriage records

Search criteria Groom Bride

First name

Last name

Place of birth

Example of ARIA

aria-label vs title

+43 666 1234 4321

+43 666 1234 4321

Non meaningful elements (aria-label)

x x

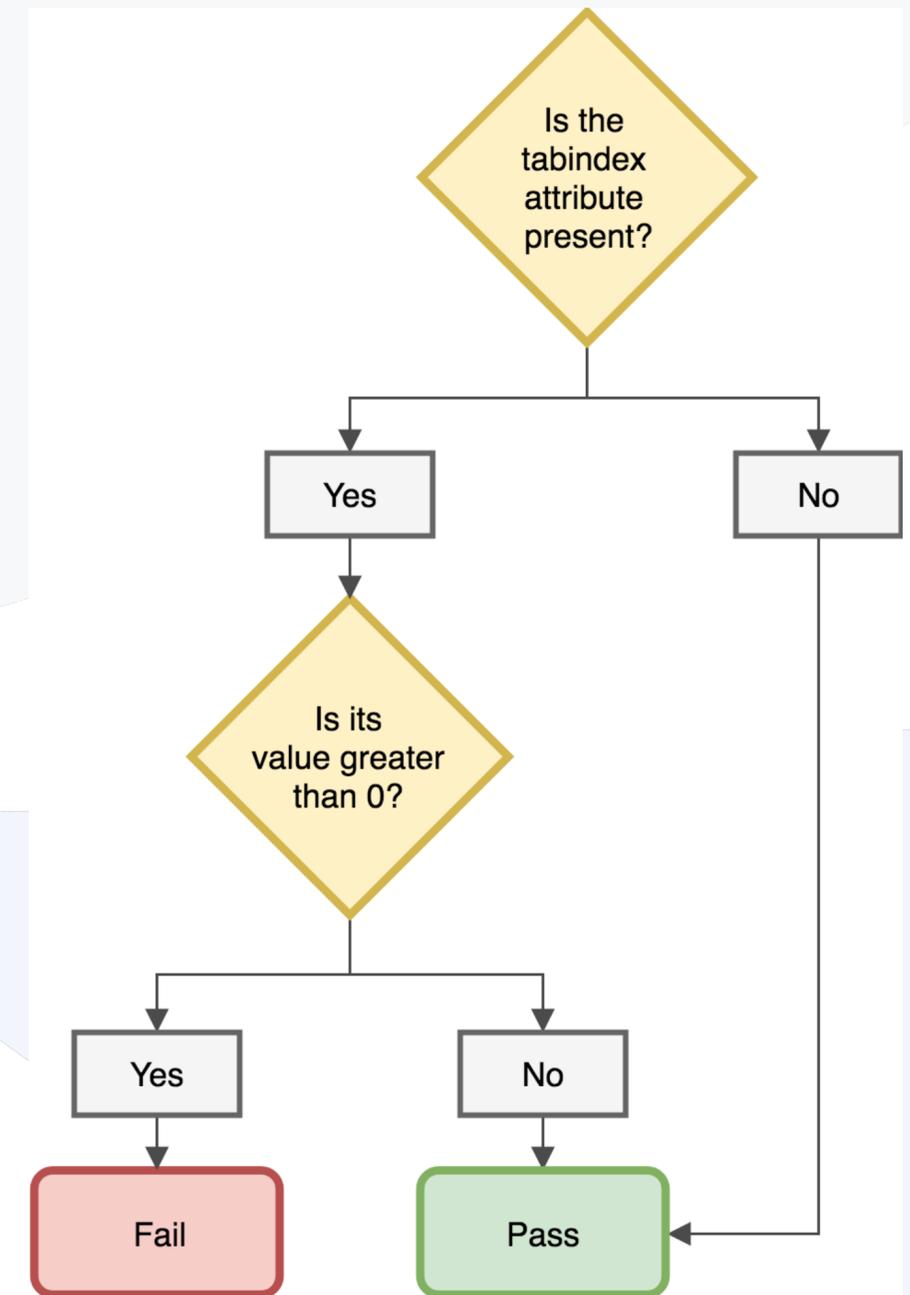
Testing Web Accessibility

How to find accessibility vulnerabilities in your website

Testing Web Accessibility

Automated a11y tests can free up your QA team from manual testing every part of your application, but they can't automatically make your site accessible.

Use automated a11y tests as one step of a larger testing process. Don't forget that only 20% to 50% of all accessibility issues can automatically be detected



I've created a simple application with 3 tiny components and a bunch of errors in the HTML

```
image.jsx — testing-web-a11y

App.js
function App() {
  return (
    <div className="App">
      <Button classes="button" text="Click me NOW" />
      <br />
      <br />
      <button role="button">CLICK ME</button>
      <br />
      <br />
      <div onClick={() => {}} />
      <div tabIndex={0} />
      <br />
      <br />
      <FakeButton classes="button" text="Click me NOW" />
      <br />
      <br />
      <a aria-label="Continue">Click me NOW</a>
      <br />
      <a aria-label="Continue" role="button">
        Click me NOW
      </a>
      <br />
      <a role="myCustomRole">Click me NOW</a>
      <a />
      <br />
      <br />
      <hr />
      <br />
      <br />
      <Image src="https://cms-assets.tutsplus.com/uploads/users/30/posts/29042/previ
      <br />
      <br />
       {
  return <button className={props.classes}>{props.text}</button>;
};

export default Button;

fakeButton.jsx
import React from "react";

const FakeButton = props => {
  return (
    <a href={props.href} className={props.classes} role="button">
      {props.text}
    </a>
  );
};

export default FakeButton;

image.jsx
import React from "react";

const Image = props => {
  return <img width="200px" src={props.src} alt={props.alt} />;
};

export default Image;
```

I've created a simple application with 3 tiny components and a bunch of errors in the HTML

```
image.jsx — testing-web-a11y

JS App.js x
19
20 function App() {
21   return (
22     <div className="App">
23       <Button classes="button" text="Click me NOW" />
24       <br />
25       <br />
26       <button role="button">CLICK ME</button>
27       <br />
28       <br />
29       <div onClick={() => {}} />
30       <div tabIndex={0} />
31       <br />
32       <br />
33       <FakeButton classes="button" text="Click me NOW" />
34       <br />
35       <br />
36       <a aria-label="Continue">Click me NOW</a>
37       <br />
38       <a aria-label="Continue" role="button">
39         Click me NOW
40       </a>
41       <br />
42       <a role="myCustomRole">Click me NOW</a>
43       <a />
44       <br />
45       <br />
46       <br />
47       <br />
48       <br />
49       <Image src="https://cms-assets.tutsplus.com/uploads/users/30/posts/29042/previ
50       <br />
51       <br />
52        {
4   return <button className={props.classes}>{props.text}</button>;
5 };
6
7 export default Button;
8

fakeButton.jsx x
1 import React from "react";
2
3 const FakeButton = props => {
4   return (
5     <a href={props.href} className={props.classes} role="button">
6       {props.text}
7     </a>
8   );
9 };
10
11 export default FakeButton;
12

image.jsx x
1 import React from "react";
2
3 const Image = props => {
4   return <img width="200px" src={props.src} alt={props.alt} />;
5 };
6
7 export default Image;
8
```

Ln 3, Col 1 Spaces: 2 UTF-8 LF JavaScript React Prettier: ✓

Testing the code

- **react-axe**

<https://github.com/dequelabs/react-axe>

- **eslint-plugin-jsx-a11y**

<https://github.com/evcohen/eslint-plugin-jsx-a11y>

- **jest-axe**

<https://github.com/nickcolley/jest-axe>

react-axe

Install it as a dev dependency

```
npm install --save-dev react-axe
```

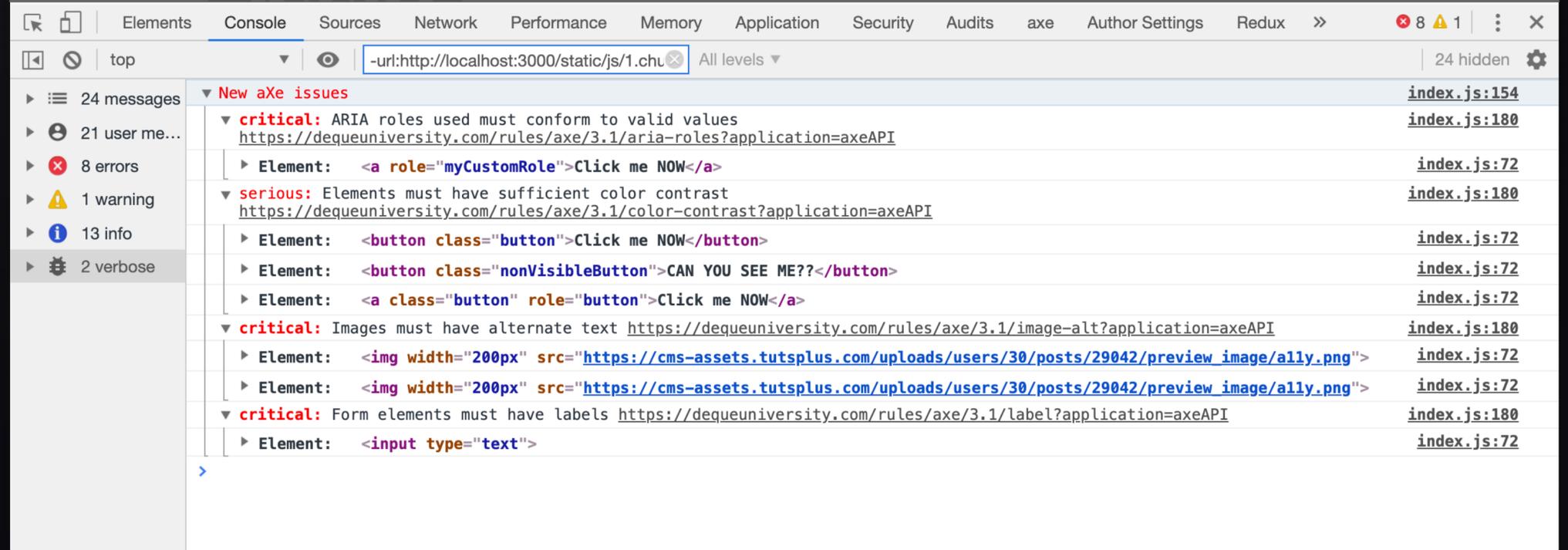
Call the exported function passing in the React and ReactDOM objects as well as a timing delay in milliseconds that will be observed between each component change and the time the analysis starts.

```
1 var React = require('react');
2 var ReactDOM = require('react-dom');
3
4
5 if (process.env.NODE_ENV !== "production") {
6   var axe = require("react-axe");
7   axe(React, ReactDOM, 1000);
8 }
```

Results will show in the Chrome DevTools console.

A severity level is also assigned for each violation. The possible levels are:

- Minor
- Moderate
- Serious
- Critical



The screenshot shows the Chrome DevTools console with the 'axe' extension installed. The console displays a list of 'New axe issues' with the following details:

Severity	Issue Description	Source
critical	ARIA roles used must conform to valid values	index.js:154
	Element: <code>Click me NOW</code>	index.js:180
serious	Elements must have sufficient color contrast	index.js:180
	Element: <code><button class="button">Click me NOW</button></code>	index.js:72
	Element: <code><button class="nonVisibleButton">CAN YOU SEE ME??</button></code>	index.js:72
	Element: <code>Click me NOW</code>	index.js:72
critical	Images must have alternate text	index.js:180
	Element: <code></code>	index.js:72
	Element: <code></code>	index.js:72
critical	Form elements must have labels	index.js:180
	Element: <code><input type="text"></code>	index.js:72

eslint-plugin-jsx-a11y

The linting issues will be displayed directly in the code editor

Configure the rules you want to use under the rules section in the .eslintrc.json file

```
.eslintrc.json — testing-web-a11y
.eslintrc.json X
.eslintrc.json > ...
Adrian Bolonio, 5 months ago | 1 author (Adrian Bolonio)
1 {
2   "plugins": ["jsx-a11y"],
3   "parserOptions": {
4     "ecmaVersion": 6,
5     "sourceType": "module",
6     "ecmaFeatures": {
7       "jsx": true,
8       "modules": true,
9       "experimentalObjectRestSpread": true
10    }
11  },
12  "extends": ["plugin:jsx-a11y/recommended"],
13  "rules": {
14    "jsx-a11y/alt-text": [
15      2,
16      {
17        "elements": ["img", "object", "area", "input[type=\"image\"]"],
18        "img": ["Image"],
19        "object": ["Object"],
20        "area": ["Area"],
21        "input[type=\"image\"]": ["InputImage"]
22      }
23    ],
24    "jsx-a11y/no-noninteractive-tabindex": [
25      "error"
```

```
App.js — testing-web-a11y
JS App.js X
src > JS App.js > ...
22 <div className="App">
23   (JSX attribute) React.DOMAttributes<HTMLDivElement>.onClick?: (event: React.MouseEvent<HTMLDivElement, MouseEvent>) => void
24
25   Static HTML elements with event handlers require a
26   role. eslint(jsxA11y/no-static-element-interactions)
27
28   Visible, non-interactive elements with click handlers must have at
29   least one keyboard listener. eslint(jsxA11y/click-events-have-key-
30   events)
31   Peek Problem Quick Fix...
32   <div onClick={() => {}} />
33   <div tabIndex={0} />
34   <br />
35   <br />
36   <FakeButton classes="button" text="Click me NOW" />
37   <br />
38   <br />
39   <a aria-label="Continue">Click me NOW</a>
40   <br />
41   <a aria-label="Continue" role="button">
42     Click me NOW
43   </a>
44   <br />
45   <a role="myCustomRole">Click me NOW</a>
46   <a />
47   <br />
```

eslint-plugin-jsx-a11y

The linting issues will be displayed directly in the code editor

Configure the rules you want to use under the rules section in the .eslintrc.json file

```
.eslintrc.json — testing-web-a11y
.eslintrc.json > ...
Adrian Bolonio, 5 months ago | 1 author (Adrian Bolonio)
1 {
2   "plugins": ["jsx-a11y"],
3   "parserOptions": {
4     "ecmaVersion": 6,
5     "sourceType": "module",
6     "ecmaFeatures": {
7       "jsx": true,
8       "modules": true,
9       "experimentalObjectRestSpread": true
10    }
11  },
12  "extends": ["plugin:jsx-a11y/recommended"],
13  "rules": {
14    "jsx-a11y/alt-text": [
15      2,
16      {
17        "elements": ["img", "object", "area", "input[type=\"image\"]"],
18        "img": ["Image"],
19        "object": ["Object"],
20        "area": ["Area"],
21        "input[type=\"image\"]": ["InputImage"]
22      }
23    ],
24    "jsx-a11y/no-noninteractive-tabindex": [
25      "error"
```

```
App.js — testing-web-a11y
JS App.js > ...
22 <div className="App">
23   (JSX attribute) React.DOMAttributes<HTMLDivElement>.onClick?: (eve
24   Static HTML elements with event handlers require a
```

The linting issues will be displayed in the terminal as well

```
testing-web-a11y — node • npm TERM_PROGRAM=Apple_Terminal NVM_CD_FLAGS= SHELL=/bin/bash — 92x49
Compiled with warnings.

./src/App.js
Line 26: The element button has an implicit role of button. Defining this explicitly is
redundant and should be avoided

jsx-a11y/no-redundant-roles
Line 39: The href attribute is required for an anchor to be keyboard accessible. Provide
a valid, navigable address as the href value. If you cannot provide an href, but still need
the element to resemble a link, use a button and change it with appropriate styles. Learn m
ore: https://github.com/evcohen/eslint-plugin-jsx-a11y/blob/master/docs/rules/anchor-is-valid
d.md jsx-a11y/anchor-is-valid
Line 41: The href attribute is required for an anchor to be keyboard accessible. Provide
a valid, navigable address as the href value. If you cannot provide an href, but still need
the element to resemble a link, use a button and change it with appropriate styles. Learn m
ore: https://github.com/evcohen/eslint-plugin-jsx-a11y/blob/master/docs/rules/anchor-is-valid
d.md jsx-a11y/anchor-is-valid
Line 45: The href attribute is required for an anchor to be keyboard accessible. Provide
a valid, navigable address as the href value. If you cannot provide an href, but still need
the element to resemble a link, use a button and change it with appropriate styles. Learn m
ore: https://github.com/evcohen/eslint-plugin-jsx-a11y/blob/master/docs/rules/anchor-is-valid
d.md jsx-a11y/anchor-is-valid
Line 45: Elements with ARIA roles must use a valid, non-abstract ARIA role
```

jest-axe

Install it as a dev dependency

```
npm install --save-dev jest-axe
```

Write an accessibility test using ReactDOMServer to render the html of your application and check for violations

```
App.test.js — testing-web-a11y
App.test.js x
src > App.test.js > ...
Adrian Bolonio, 5 months ago | 1 author (Adrian Bolonio)
1 import { axe, toHaveNoViolations } from "jest-axe";
2 import React from "react";
3 import ReactDOMServer from "react-dom/server";
4 import App from "./App";
5
6 expect.extend(toHaveNoViolations);
7
8 it("should demonstrate this matcher`s usage with react", async () => {
9   const html = ReactDOMServer.renderToString(<App />);
10  const results = await axe(html);
11  expect(results).toHaveNoViolations();
12 });
13
```

The results of the test will be displayed in the terminal

```
testing-web-a11y — node • npm TERM_PROGRAM=Apple_Terminal NVM_CD_FLAGS= SHELL=/bin/bash — 119x52
FAIL src/App.test.js
  ✕ should demonstrate this matcher`s usage with react (605ms)
    • should demonstrate this matcher`s usage with react
      expect(received).toHaveNoViolations(expected)
      Expected the HTML found at $('a[role="myCustomRole"]') to have no violations:
      <a role="myCustomRole">Click me NOW</a>
      Received:
      "ARIA roles used must conform to valid values (aria-roles)"
      Try fixing it with this help: https://dequeuniversity.com/rules/axe/3.1/aria-roles?application=axeAPI
      _____
      Expected the HTML found at $('img:nth-child(28)') to have no violations:
      ') to have no violations:
      <input type="text">
      Expected the HTML found at $('label:nth-child(40) > input') to have no violations:
      <input type="text">
      Received:
      "Form elements must have labels (label)"
```

Testing the DOM

- **axe-cli**

<https://github.com/dequelabs/axe-cli>

- **pa11y**

<http://pa11y.org/>

- **Lighthouse**

<https://developers.google.com/web/tools/lighthouse/>

axe-cli

Install axe-cli globally

```
npm install axe-cli -g
```

You can now run the axe command in your CLI, followed by the URL of the page you wish to test

```
adrianbolonio@Adrians-MBP testing-web-a11y (master) $ axe https://stackoverflow.com/
Running axe-core 3.1.2 in chrome-headless

Testing https://stackoverflow.com/ ... please wait, this may take a minute.

Violation of "aria-valid-attr-value" with 1 occurrences!
Ensures all ARIA attributes have valid values. Correct invalid elements at:
- .wmx8
For details, see: https://dequeuniversity.com/rules/axe/3.1/aria-valid-attr-value

Violation of "color-contrast" with 7 occurrences!
Ensures the contrast between foreground and background colors meets WCAG 2 AA contrast ratio thresholds. Correct invalid elements at:
- .is-selected
- .btn-topbar-clear
- .s-btn__primary
- .js-link
- .bc-orange-2
- .sm\:m10
- .mt-auto.mb24
For details, see: https://dequeuniversity.com/rules/axe/3.1/color-contrast

Violation of "heading-order" with 3 occurrences!
Ensures the order of headings is semantically correct. Correct invalid elements at:
- .mt64.h\:bs-lg.md\:mt0:nth-child(1) > .fs-title.p-ff-roboto-slab
- .bs-sm.w100.md\:mb32:nth-child(1) > .pb16.bb.mln16 > .mb8.fs-title
- .site-footer--col__visible.js-footer-col[data-name="default"]:nth-child(1) > h5
For details, see: https://dequeuniversity.com/rules/axe/3.1/heading-order

Violation of "html-has-lang" with 1 occurrences!
Ensures every HTML document has a lang attribute. Correct invalid elements at:
- html
For details, see: https://dequeuniversity.com/rules/axe/3.1/html-has-lang

Violation of "image-alt" with 24 occurrences!
Ensures <img> elements have alternate text or a role of none or presentation. Correct invalid elements at:
- .mt64.h\:bs-lg.md\:mt0:nth-child(1) > .mb16:nth-child(1)
- .ml64 > .mb16:nth-child(1)
- .mt64.h\:bs-lg.md\:mt0:nth-child(3) > .mb16:nth-child(1)
- .grid__center.wmx4.lg\:fd-column:nth-child(1) > .lg\:mb16.md\:mb0.mr16 > img[width="\37 6"]
- .ml48.mr48.grid__center > .lg\:mb16.md\:mb0.mr16 > img[width="\37 6"]
- .grid__center.wmx4.lg\:fd-column:nth-child(3) > .lg\:mb16.md\:mb0.mr16 > img[width="\37 6"]
- .sm\:mb64 > .grid--cell:nth-child(1) > .mb6[width="\37 6"]
- .fd-column.grid:nth-child(2) > .grid--cell:nth-child(1) > .mb6[width="\37 6"]
- .is-active > .md\:mb8.md\:mr0.mr8 > img[width="\34 6"]
- .tour--controls.md\:w100.grid--cell3:nth-child(1) > label:nth-child(2) > .md\:mb8.md\:mr0.mr8 > img[width="\34 6"]
- .tour--controls.md\:w100.grid--cell3:nth-child(1) > label:nth-child(3) > .md\:mb8.md\:mr0.mr8 > img[width="\34 6"]
- .tour--controls__right > label:nth-child(1) > .md\:mb8.md\:mr0.mr8 > img[width="\34 6"]
- .tour--controls__right > label:nth-child(2) > .md\:mb8.md\:mr0.mr8 > img[width="\34 6"]
- .tour--controls__right > label:nth-child(3) > .md\:mb8.md\:mr0.mr8 > img[width="\34 6"]
```

pa11y

Install pa11y globally

```
npm install -g pa11y
```

You can now run the pa11y command in your CLI, followed by the URL of the page you wish to test

```
adrianbolonio@Adrians-MBP testing-web-a11y (master) $ pa11y https://stackoverflow.com/
(node:50833) ExperimentalWarning: The fs.promises API is experimental

Welcome to Pa11y

> Running Pa11y on URL https://stackoverflow.com/

Results for URL: https://stackoverflow.com/

• Error: The html element should have a lang or xml:lang attribute which describes the language of the document.
  └─ WCAG2AA.Principle3.Guideline3_1.3_1_1.H57.2
     └─ html
        └─ <html class="html__responsive html__unpinned-leftnav"><head> <title>Stack Ov...</html>

• Error: Anchor element found with a valid href attribute, but no link content has been supplied.
  └─ WCAG2AA.Principle4.Guideline4_1.4_1_2.H91.A.NoContent
     └─ html > body > header > div > div:nth-child(1) > a:nth-child(1)
        └─ <a href="#" class="left-sidebar-toggle p0 ai-center jc-center js-left-sidebar-toggle" role="menuitem" aria-haspopup="true" aria-controls="left-sidebar" aria-expanded="false"><span class="ps-relative"></span...</a>

• Error: This form does not contain a submit button, which creates issues for those who cannot submit the form using the keyboard. Submit buttons are INPUT elements with type attribute "submit" or "image", or BUTTON elements with type "submit" or omitted/invalid.
  └─ WCAG2AA.Principle3.Guideline3_2.3_2_2.H32.2
     └─ #search
        └─ <form id="search" role="search" action="/search" method="get" class="grid--cell fl-grow1 searchbar px12 js-searchbar " autocomplete="off"> <div class...</form>

• Error: Anchor element found with no link content and no name and/or ID attribute.
  └─ WCAG2AA.Principle4.Guideline4_1.4_1_2.H91.A.EmptyNoId
     └─ #js-gdpr-consent-banner > div > a
        └─ <a class="grid--cell fc-white js-notice-close" aria-label="notice-dismiss"> <svg aria-hidd...</a>

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 4.16:1. Recommendation: change background to #353a3f.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #announcement-banner > div > div > a
        └─ <a target="_blank" class="s-link js-link js-gps-track fw-bold fc-blue-400 " href="https://stackoverflow.blog/2019/11/13/were-rewarding-the-question-askers/" data-ga="["&quot;rep-recalc-2019&quot;,&quot;Announcement Banner&quot;,&quot;https://stackover...

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 2.64:1. Recommendation: change text colour to #c35600.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #content > div:nth-child(1) > div > a:nth-child(1)
        └─ <a class="grid--cell bc-orange-2 ba fc-orange-400 bg-white py12 px24 bar-sm js-scroll-link js-gps-track" data-gps-track="product_homepage.hero.click({ cta: 'For developers' })" href="#for-developers" data-ga="["&quot;home page&quot;,&quot;for develope...

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 2.64:1. Recommendation: change background to #c35600.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #content > div:nth-child(1) > div > a:nth-child(2)
```

pa11y

Install pa11y globally



```
npm install -g pa11y
```

You can now run the pa11y command in your CLI, followed by the URL of the page you wish to test

```
adrianbolonio@Adrians-MBP testing-web-a11y (master) $ pa11y https://stackoverflow.com/
(node:50833) ExperimentalWarning: The fs.promises API is experimental

Welcome to Pa11y

> Running Pa11y on URL https://stackoverflow.com/

Results for URL: https://stackoverflow.com/

• Error: The html element should have a lang or xml:lang attribute.
  └─ WCAG2AA.Principle3.Guideline3_1.3_1_1.H57.2
     └─ html
        └─ <html class="html_responsive html_unpinned-left">

• Error: Anchor element found with a valid href attribute but no link content and no name and/or ID attribute.
  └─ WCAG2AA.Principle4.Guideline4_1.4_1_2.H91.A.NoContent
     └─ html > body > header > div > div:nth-child(1) > a
        └─ <a href="#" class="left-sidebar-toggle p8 ai-center" data-ga="["&quot;home page&quot;,&quot;announcement banner&quot;]&quot; announcement banner"></a>

• Error: This form does not contain a submit button, which may be necessary for user interaction.
  └─ WCAG2AA.Principle3.Guideline3_2.3_2_2.H32.2
     └─ #search
        └─ <form id="search" role="search" action="/search">

• Error: Anchor element found with no link content and no name and/or ID attribute.
  └─ WCAG2AA.Principle4.Guideline4_1.4_1_2.H91.A.EmptyNoId
     └─ #js-gdpr-consent-banner > div > a
        └─ <a class="grid-cell fc-white js-notice-close" href="#">

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 2.95:1. Recommendation: change background to #d93d21.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #root > div > button:nth-child(1)
        └─ <button class="button">Click me NOW</button>

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 1.05:1. Recommendation: change background to #5d7c7c.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #root > div > button:nth-child(7)
        └─ <button class="nonVisibleButton">CAN YOU SEE ME??</button>

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 2.95:1. Recommendation: change background to #d93d21.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #content > div:nth-child(1) > div > a:nth-child(14)
        └─ <a class="button" role="button">Click me NOW</a>

• Error: Anchor element found with no link content and no name and/or ID attribute.
  └─ WCAG2AA.Principle4.Guideline4_1.4_1_2.H91.A.EmptyNoId
     └─ #root > div > a:nth-child(22)
```

You can run the pa11y command in your CLI for localhost as well

```
adrianbolonio@Adrians-MacBook-Pro testing-web-a11y (master) $ pa11y http://localhost:3000/
(node:19277) ExperimentalWarning: The fs.promises API is experimental

Welcome to Pa11y

> Running Pa11y on URL http://localhost:3000/

Results for URL: http://localhost:3000/

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 2.95:1. Recommendation: change background to #d93d21.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #root > div > button:nth-child(1)
        └─ <button class="button">Click me NOW</button>

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 1.05:1. Recommendation: change background to #5d7c7c.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #root > div > button:nth-child(7)
        └─ <button class="nonVisibleButton">CAN YOU SEE ME??</button>

• Error: This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contrast ratio of 2.95:1. Recommendation: change background to #d93d21.
  └─ WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail
     └─ #content > div > a:nth-child(14)
        └─ <a class="button" role="button">Click me NOW</a>

• Error: Anchor element found with no link content and no name and/or ID attribute.
  └─ WCAG2AA.Principle4.Guideline4_1.4_1_2.H91.A.EmptyNoId
     └─ #root > div > a:nth-child(22)
```

pa11y-ci

Install pa11y-ci globally

```
npm install -g pa11y-ci
```

By default, Pa11y CI looks for a config file (.json) in the current working directory. Each URL in your config file can be an object and specify pa11y configurations which override the defaults too.

```
.pa11y-ci.json ×
.pa11y-ci.json > ...
You, a few seconds ago | 2 authors (You and others)
1 {
2   "urls": [
3     "https://stackoverflow.com/",
4     {
5       "url": "https://stackoverflow.com/",
6       "timeout": 50000,
7       "screenCapture": "captures-stackoverflow.png"
8     },
9     {
10      "url": "https://stackoverflow.com/",
11      "timeout": 50000,
12      "actions": [
13        "screen capture captures-stackoverflow-home.png",
14        "navigate to https://stackoverflow.com/questions",
15        "screen capture captures-stackoverflow-questions.png"
16      ]
17    }
18  ]
19 }
20
```

```
adrianbolonio@Adrians-MBP testing-web-a11y (master) $ pa11y-ci
(node:50843) ExperimentalWarning: The fs.promises API is experimental
Running Pa11y on 3 URLs:
> https://stackoverflow.com/ - 82 errors
> https://stackoverflow.com/ - 82 errors
> https://stackoverflow.com/ - 392 errors

Errors in https://stackoverflow.com/:

• The html element should have a lang or xml:lang attribute which describes the language of the document.
(html)
<html class="html__responsive"><head> <title>Newest Q...</html>

• Anchor element found with a valid href attribute, but no link content has been supplied.
(html > body > header > div > div:nth-child(1) > a:nth-child(1))
<a href="#" class="left-sidebar-toggle p0 ai-center jc-center js-left-sidebar-toggle" role="menuitem" aria-haspopup="true" aria-controls="left-sideba
class="ps-relative"></spa...</a>

• This form does not contain a submit button, which creates issues for those who cannot submit the form using the keyboard. Submit buttons are INPUT el
"submit" or "image", or BUTTON elements with type "submit" or omitted/invalid.
(#search)
<form id="search" role="search" action="/search" method="get" class="grid--cell fl-grow1 searchbar px12 js-searchbar " autocomplete="off"> <div class=

• Anchor element found with no link content and no name and/or ID attribute.
(#js-gdpr-consent-banner > div > a)
<a class="grid--cell fc-white js-notice-close" aria-label="notice-dismiss"> <svg aria-hidd...</a>

• This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contras
Recommendation: change background to #353a3f.
(#announcement-banner > div > div > a)
<a target="_blank" class="s-link js-link js-gps-track fw-bold fc-blue-400 " href="https://stackoverflow.blog/2019/11/13/were-rewarding-the-question-a
data-ga="[&quot;rep-recalc-2019&quot;,&quot;Announcement Banner&quot;,&quot;https://stackover...

• This element has insufficient contrast at this conformance level. Expected a contrast ratio of at least 4.5:1, but text in this element has a contras
change text colour to #fefeef.
(#nav-questions > span)
<span class="--link--channel-name">Stack Overflow</span>
```

pa11y-ci

Install pa11y-ci globally

By default, Pa11y CI looks for a config file (.json) in the current working directory. Each URL in your config file can be an object and specify pa11y configurations which override the defaults too.

```
npm install -g pa11y-ci
```

```
.pa11y-ci.json X
.pa11y-ci.json > ...
You, a few seconds ago | 2 authors (You and others)
1 {
2   "urls": [
3     "https://stackoverflow.com/",
4     {
5       "url": "https://stackoverflow.com/",
6       "timeout": 50000,
7       "screenCapture": "captures-stackoverflow.png"
8     },
9     {
10      "url": "https://stackoverflow.com/",
11      "timeout": 50000,
12      "actions": [
13        "screen capture captures-stackoverflow-home.png",
14        "navigate to https://stackoverflow.com/questions",
15        "screen capture captures-stackoverflow-questions.png"
16      ]
17    }
18  ]
19 }
20
```

The screenshot displays a development environment with three main components:

- Terminal Window:** Shows the execution of pa11y-ci on three URLs. The output indicates 82 errors for the first two URLs and 392 errors for the third. A detailed error log for the first URL is visible, listing accessibility issues such as missing lang attributes, invalid href attributes, and insufficient contrast.
- Explorer View:** Shows a file tree for a project named 'TESTING-WEB-A11Y'. The tree includes folders like 'node_modules', 'public', and 'src', and files like 'App.css', 'App.js', 'App.test.js', 'index.css', 'index.js', 'logo.svg', and 'serviceWorker.js'. The '.pa11y-ci.json' file is highlighted in blue.
- Code Editor:** Shows the content of the '.pa11y-ci.json' file, which is a JSON array of objects. Each object represents a URL to be tested, with optional pa11y configuration options like 'timeout' and 'screenCapture'.

Lighthouse

Install lighthouse globally

```
npm install -g lighthouse
```

You can now run the lighthouse command in the terminal (with `--view`) and it will immediately open the generated html report.

Lighthouse Report

File | /Users/adrianbolonio/Do... ☆

Twitter LinkedIn YouTube Banco Wien willhaben Noticias Dev Design

Lighthouse 4.3.1 <https://stackoverflow.com/>

75 Performance 63 Accessibility 79 Best Practices 87 SEO PWA Progressive Web App

Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

63

ARIA

These are opportunities to improve the usage of ARIA in your application which may enhance the experience for users of assistive technology, like a screen reader.

- [aria-*] attributes do not have valid values ▲

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. [Learn more.](#)

Failing Elements

```
<div class="wmx8 mx-auto grid jc-center" role="alertdialog" aria-describedby="notice-message">
```

Contrast

These are opportunities to improve the legibility of your content.

- Background and foreground colors do not have a sufficient contrast ratio. ▲

Internationalization and localization

These are opportunities to improve the interpretation of your content by users in different locales.

- <html> element does not have a [lang] attribute ▲

Names and labels

These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users

```
testing-web-a11y -- -bash -- 92x49
status Evaluating: Interactive controls are keyboard focusable +1ms
status Evaluating: Headings don't skip levels +0ms
status Evaluating: Interactive elements indicate their purpose and state +0ms
status Evaluating: The page has a logical tab order +0ms
status Evaluating: The user's focus is directed to new content added to the page +0ms
status Evaluating: Offscreen content is hidden from assistive technology +0ms
status Evaluating: HTML5 landmark elements are used to improve navigation +0ms
status Evaluating: Visual order on the page follows DOM order +0ms
status Evaluating: Uses efficient cache policy on static assets +1ms
status Evaluating: Avoids enormous network payloads +6ms
status Evaluating: Defer offscreen images +3ms
status Evaluating: Eliminate render-blocking resources +45ms
status Evaluating: Minify CSS +24ms
status Evaluating: Minify JavaScript +108ms
status Evaluating: Remove unused CSS +86ms
status Evaluating: Serve images in next-gen formats +20ms
status Evaluating: Efficiently encode images +14ms
status Evaluating: Enable text compression +9ms
status Evaluating: Properly size images +16ms
status Evaluating: Use video formats for animated content +13ms
status Evaluating: Avoids Application Cache +11ms
status Evaluating: Page has the HTML doctype +1ms
status Evaluating: Avoids an excessive DOM size +3ms
status Evaluating: Links to cross-origin destinations are safe +5ms
status Evaluating: Avoids requesting the geolocation permission on page load +2ms
status Evaluating: Avoids `document.write()` +1ms
status Evaluating: Avoids front-end JavaScript libraries with known security vulnerabilities +0ms
status Evaluating: Detected JavaScript libraries +9ms
status Evaluating: Avoids requesting the notification permission on page load +1ms
status Evaluating: Allows users to paste into password fields +1ms
status Evaluating: Uses HTTP/2 for its own resources +1ms
status Evaluating: Uses passive listeners to improve scrolling performance +2ms
status Evaluating: Document has a meta description +0ms
status Evaluating: Page has successful HTTP status code +1ms
status Evaluating: Document uses legible font sizes +0ms
status Evaluating: Links have descriptive text +3ms
status Evaluating: Page isn't blocked from indexing +9ms
status Evaluating: robots.txt is valid +17ms
status Evaluating: Tap targets are sized appropriately +2ms
status Evaluating: Document has a valid `hreflang` +10ms
status Evaluating: Document avoids plugins +2ms
status Evaluating: Document has a valid `rel=canonical` +2ms
status Evaluating: Structured data is valid +1ms
status Generating results... +0ms
Printer html output written to /Users/adrianbolonio/Documents/workspace/testing-web-a11y/s
tackoverflow.com_2019-11-14_16-41-59.report.html +145ms
ChromeLauncher Killing Chrome instance 50686 +11ms
adrianbolonio@Adrians-MBP testing-web-a11y (master) $
```

Testing the DOM

Main feature

- **axe-cli**

Direct link from terminal to documentation at dequeuniversity

- **pa11y**

pa11y-ci allows to include several urls with different config options that allows you to perform actions in the website

- **Lighthouse**

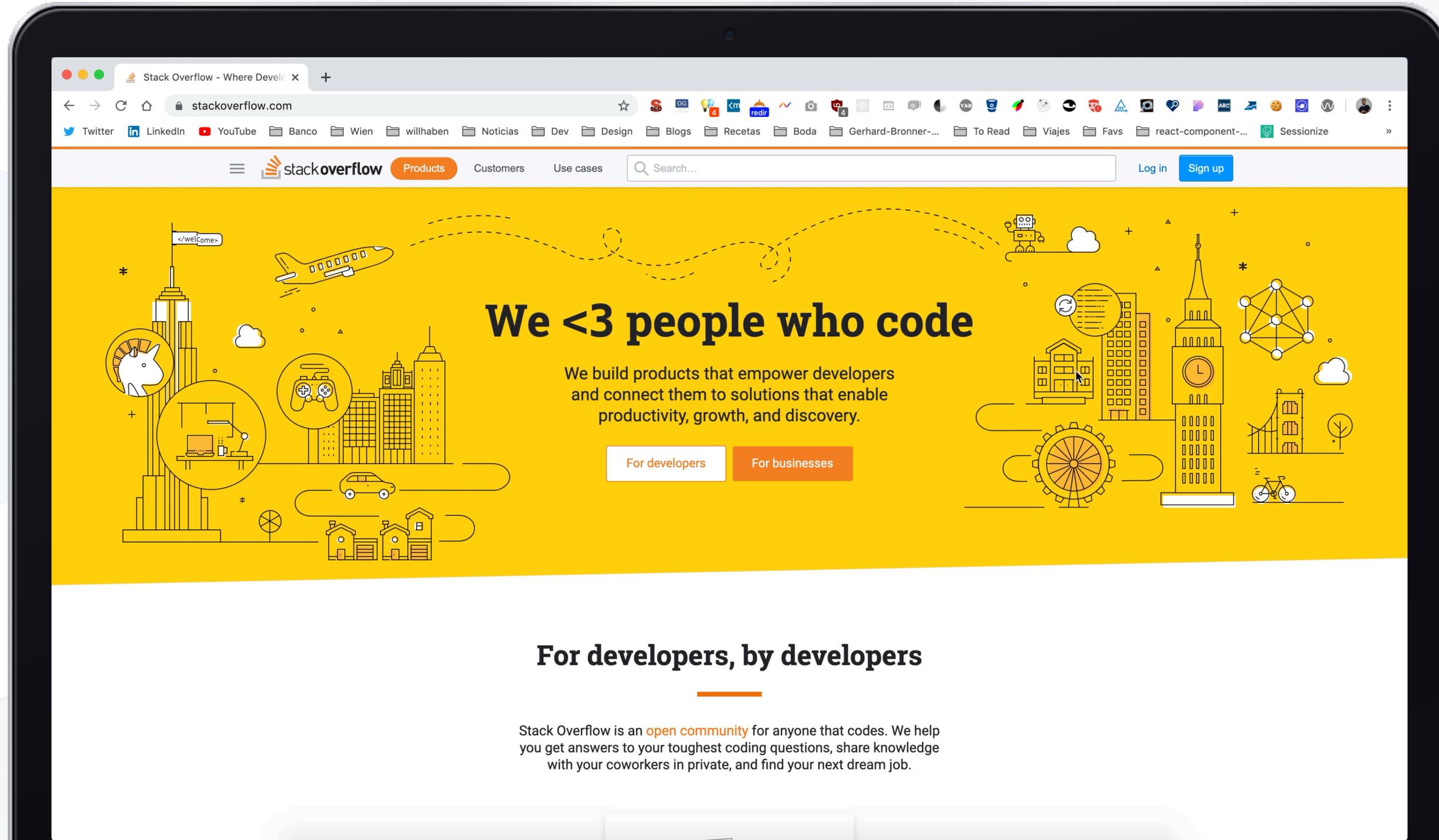
At the end of the analysis, the tools will generate a final report, either HTML or JSON/CSV the can be use for reporting purposes

Testing in the browser

- **axe chrome extension** <https://www.deque.com/axe/>
- **ARC Toolkit** <https://www.paciellogroup.com/toolkit/>
- **Accessibility Insights** <https://accessibilityinsights.io/>
- **WAVE** <https://wave.webaim.org/>
- **NoCoffee** <https://accessgarage.wordpress.com/>

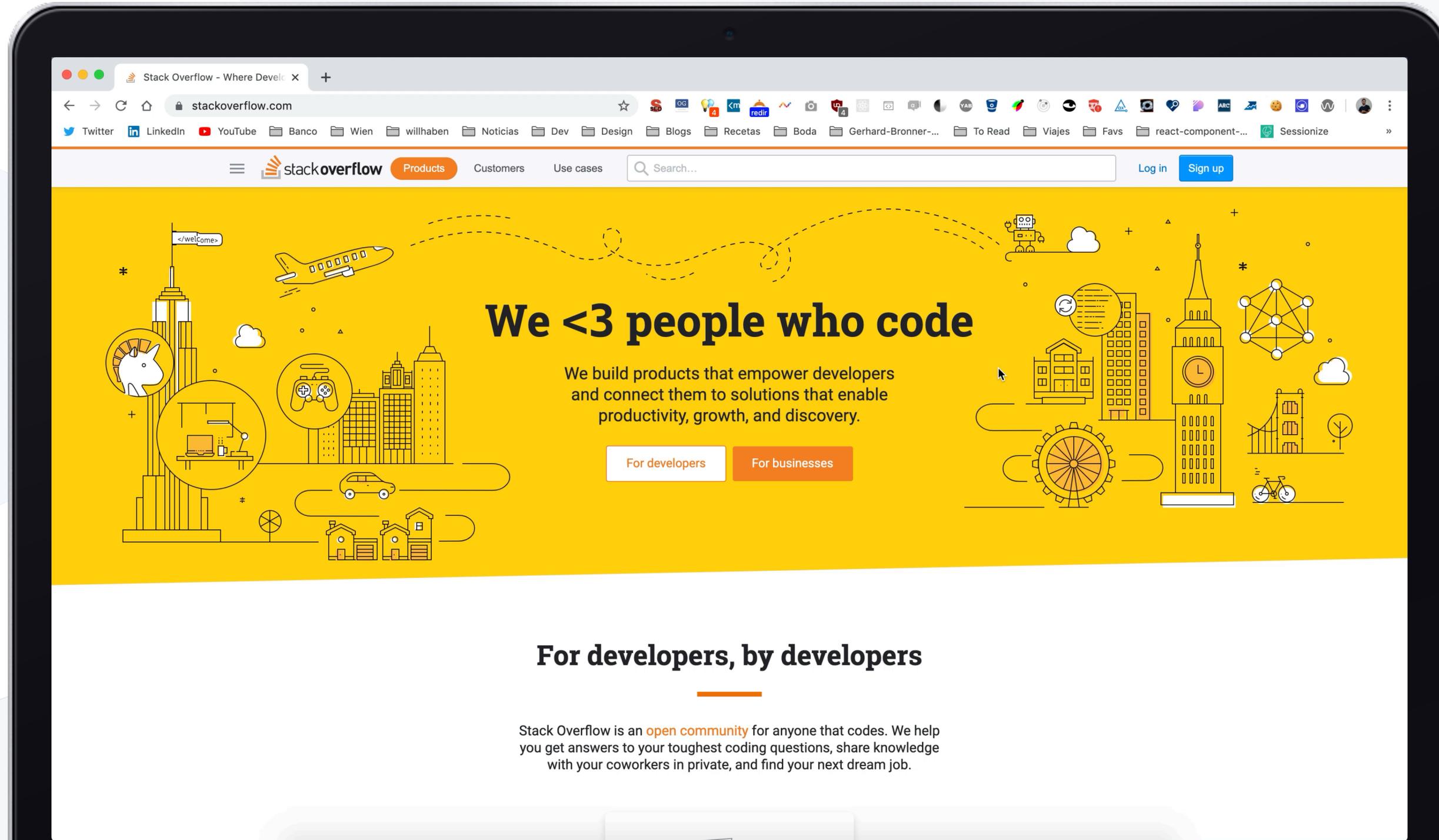
axe chrome extension

<https://www.deque.com/axe/>



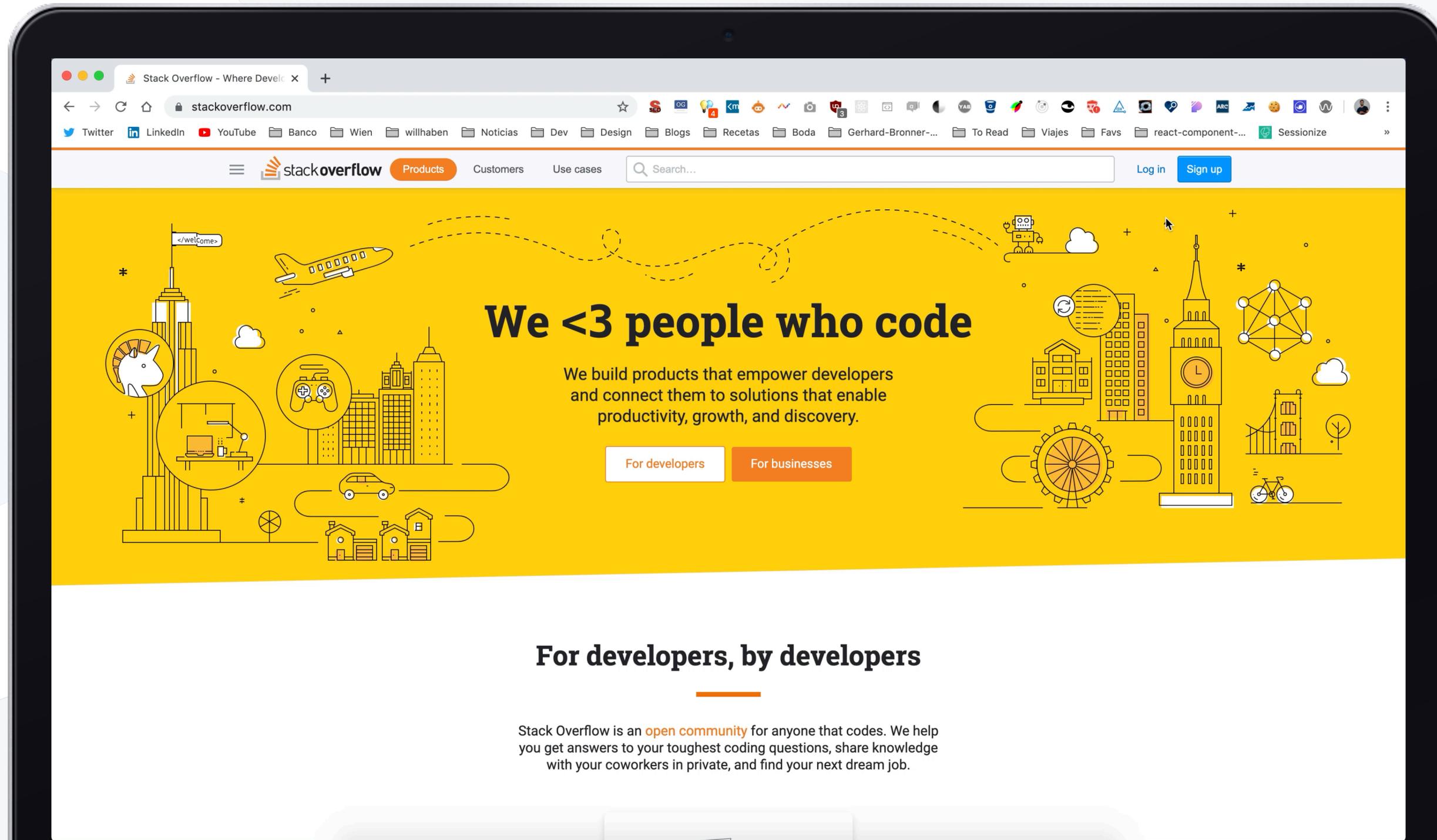
ARC Toolkit

<https://www.paciellogroup.com/toolkit/>



Accessibility Insights

<https://accessibilityinsights.io/>



We <3 people who code

We build products that empower developers and connect them to solutions that enable productivity, growth, and discovery.

For developers

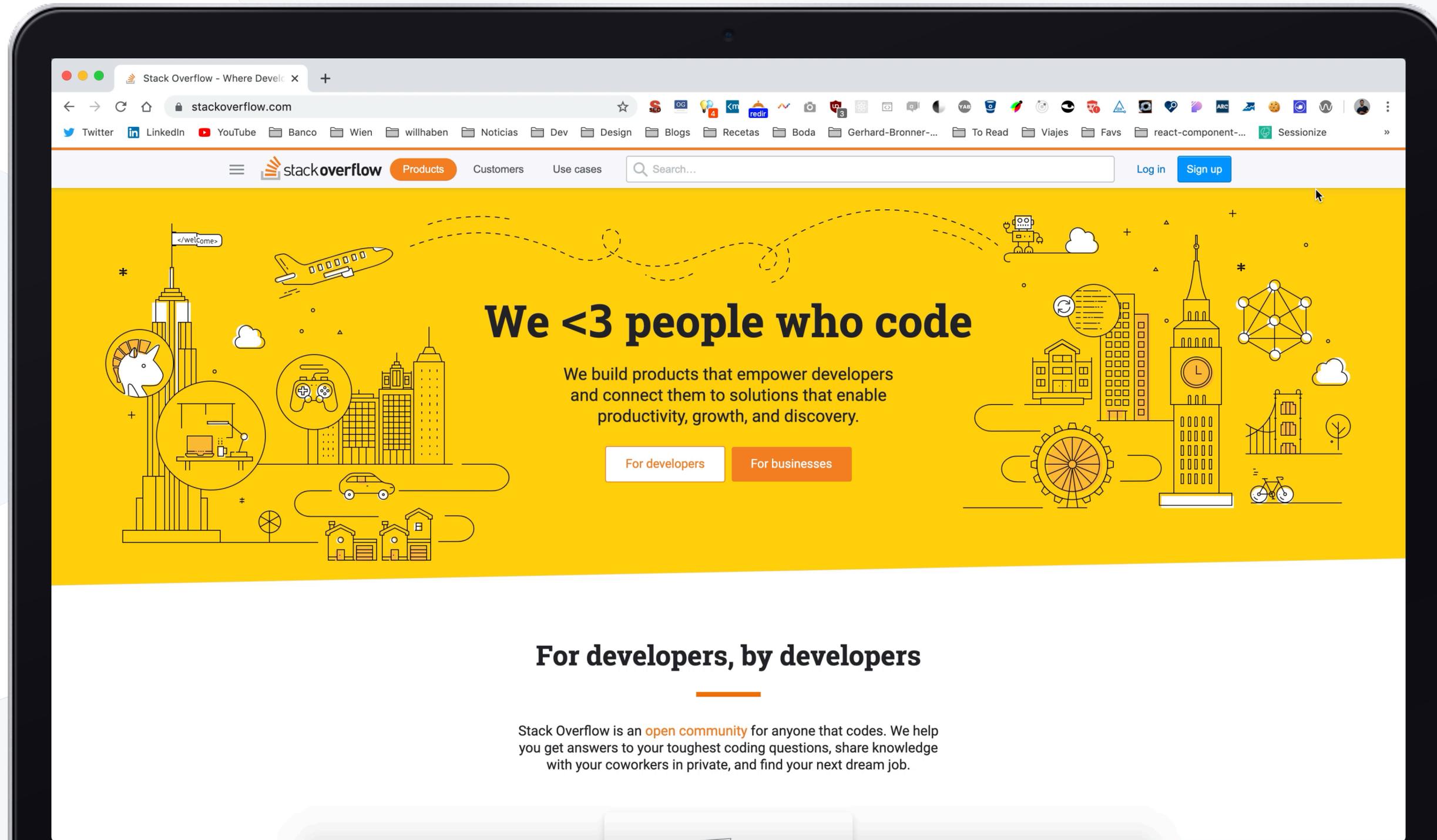
For businesses

For developers, by developers

Stack Overflow is an **open community** for anyone that codes. We help you get answers to your toughest coding questions, share knowledge with your coworkers in private, and find your next dream job.

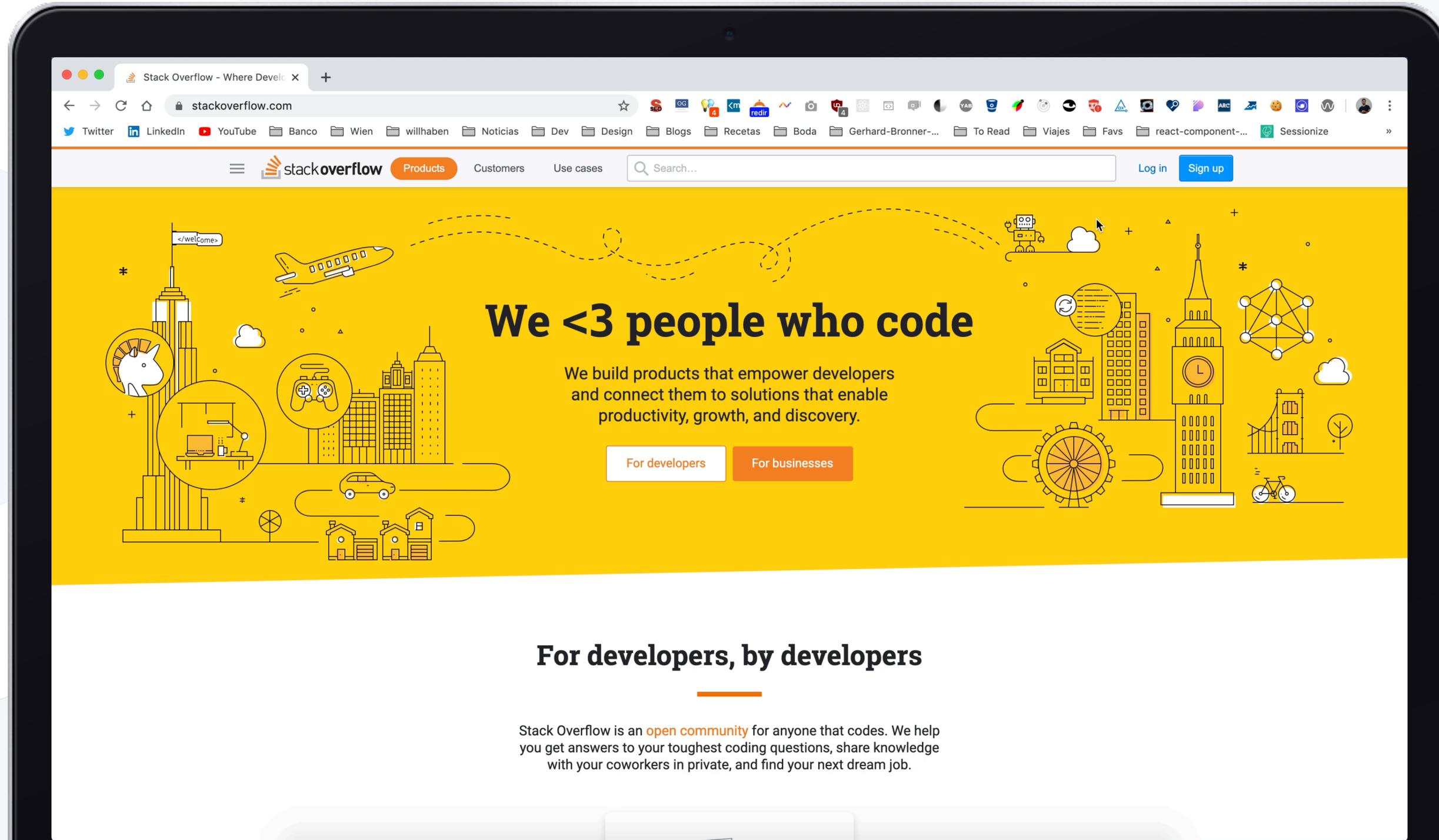
WAVE (web accessibility evaluation tool)

<https://wave.webaim.org/>



NoCoffee - Vision Simulator

<https://accessgarage.wordpress.com/>



Testing the code

- **react-axe**
 - **eslint-plugin-jsx-a11y**
 - **jest-axe**
-

Testing the DOM

- **axe-cli**
 - **pa11y**
 - **Lighthouse**
-

Testing in the browser

- **axe**
- **ARC Toolkit**
- **Accessibility Insights**
- **NoCoffee**
- **WAVE**

<https://github.com/bolonio/testing-web-a11y>

```
image.jsx — testing-web-a11y

JS App.js x
19
20 function App() {
21   return (
22     <div className="App">
23       <Button classes="button" text="Click me NOW" />
24       <br />
25       <br />
26       <button role="button">CLICK ME</button>
27       <br />
28       <br />
29       <div onClick={() => {}} />
30       <div tabIndex={0} />
31       <br />
32       <br />
33       <FakeButton classes="button" text="Click me NOW" />
34       <br />
35       <br />
36       <a aria-label="Continue">Click me NOW</a>
37       <br />
38       <a aria-label="Continue" role="button">
39         Click me NOW
40       </a>
41       <br />
42       <a role="myCustomRole">Click me NOW</a>
43       <a />
44       <br />
45       <br />
46       <hr />
47       <br />
48       <br />
49       <Image src="https://cms-assets.tutsplus.com/uploads/users/30/posts/29042/previ
50       <br />
51       <br />
52        {
4   return <button className={props.classes}>{props.text}</button>;
5 };
6
7 export default Button;
8

fakeButton.jsx x
1 import React from "react";
2
3 const FakeButton = props => {
4   return (
5     <a href={props.href} className={props.classes} role="button">
6       {props.text}
7     </a>
8   );
9 };
10
11 export default FakeButton;
12

image.jsx x
1 import React from "react";
2
3 const Image = props => {
4   return <img width="200px" src={props.src} alt={props.alt} />;
5 };
6
7 export default Image;
8
```

How does your website “sound” like?

How web accessibility can make impaired people use your website

Adrián Bolonio

 @bolonio

Testing Web Accessibility

How to find accessibility vulnerabilities in your website

Adrián Bolonio

 @bolonio

<https://speakerdeck.com/bolonio>

“It’s not just about disabled users being able to access your website, it’s about **everyone being able to access your website.”**

Trenton Moss, Owner of Webcredible Consultancy Firm, UK

Accessibility is NOT a feature

Thanks

Adrián Bolonio

 **@bolonio**