

# Web Accessibility FOR **NEWBIES**

Karina Chow  
July 15, 2021

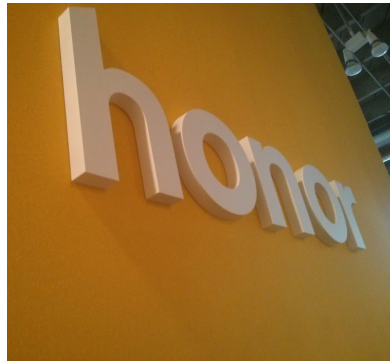


# Questions we will answer today!

- What does it mean to be “accessible”?
- What is an “ADA Lawsuit”?
- Why should we care about accessibility?
- How can we evaluate the accessibility of our websites?
- What can we do to make them more accessible?
- Where else can I look to find more information?

# Hello World! I'm Karina.

- I'm a Bay Area Native! I was born and raised in Mountain View
- I'm a 3rd generation "tech worker"
- I went to Carnegie Mellon University and got a B.S. in CS & HCI
- I've worked as a frontend/software engineer at companies such as Adobe, Yammer (now Microsoft), Honor, and Patreon
- I've had greenish hair for 11 years and my socials say "Screw the rules, I have green hair"
- I'm currently self-employed, and yes, my company name is KACHOW!



**OK Karina. That's nice.**

But why are **you** talking to us about  
**accessibility** today?

**Because it's important!**

OK, but why are **you** talking to us about  
accessibility today?

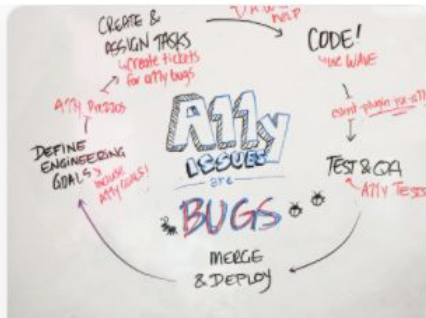
# I've written some stuff



levelup.gitconnected.com

## The Developer's Guide to Web Accessibility Auditing and Tooling

A high-level view of tooling available to audit our products for accessibili...



levelup.gitconnected.com

## 5 Ideas on How to Incorporate Web Accessibility into your Dev Workflow

Thoughts and philosophies for how to set up your team's workflow to tackl...



uxdesign.cc

## How to Convince Company Leadership to Care About Accessibility

Over 11,000 ADA lawsuits were filed in the US in 2019 and 26% of American...



levelup.gitconnected.com

## Web Accessibility for Newbies

Introductory information and advice on how to think about and approach web accessibility





I am **NOT** an expert.



What does it mean to be  
“accessible” anyway?

To be accessible is to be  
POUR

Wait what?

Accessible places and products should be

**P**erceivable

**O**perable

**U**nderstandable

**R**obust

a11y stands for  
A C C E S S I B I L I T Y  
└──────── 11 characters ─────────┘

Source: [The A11Y Project](#)

“ally”

“ally”  
That’s annoyingly cute.



Right?

OK, but why should we  
care about “ally”?

# Why?

## It's the law.

---

The **Americans with Disabilities Act (ADA)** Title III prohibits discrimination on the basis of disability

*"[it] prohibits discrimination on the basis of disabilities in places of public accommodations, commercial facilities, and private entities that offer certain examination and courses related to educational and occupational certification"*

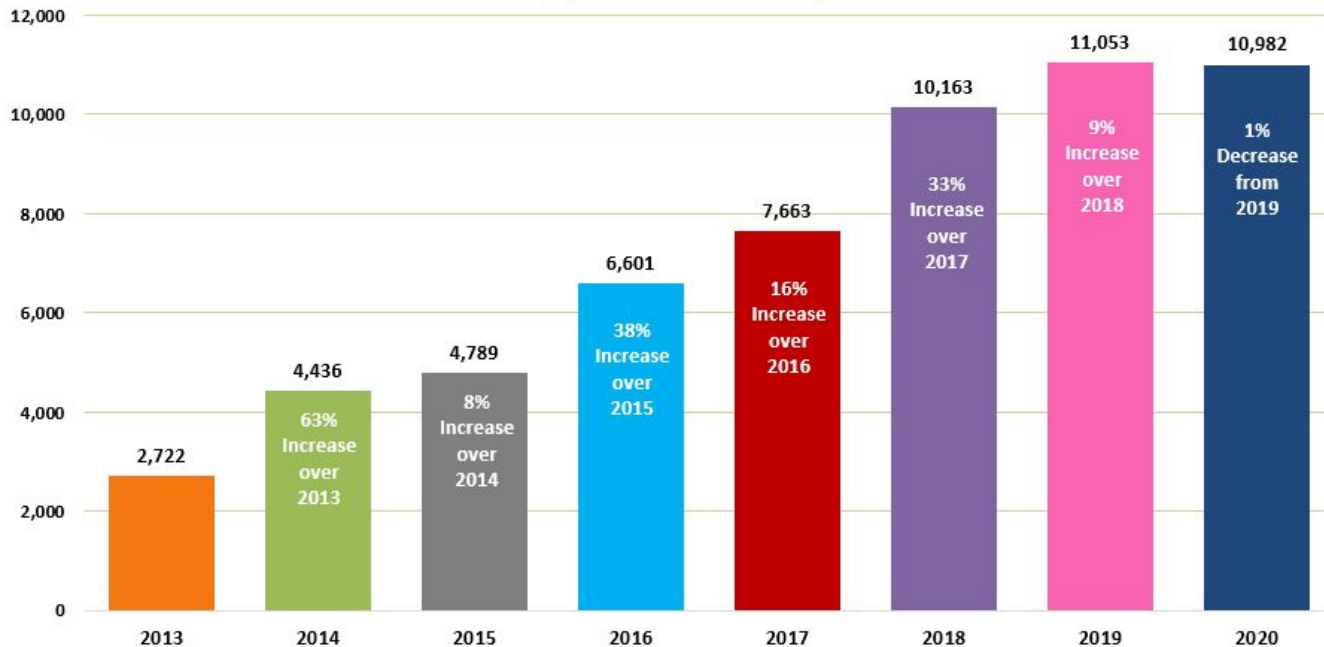
Other countries have similar laws

# Why?

## It's the law.

Turns out, you can get sued  
for breaking the law!


Total Number of ADA Title III Federal Lawsuits Filed Each Year  
January 1, 2013 - December 31, 2020




Source: [Seyfarth](#)



# Why?



A blind customer could not order his pizza through the Domino's website. He sued and won.



Search quotes, news & videos 

SIGN IN


MARKETSBUSINESSINVESTINGTECHPOLITICS  
CNBC TVWATCHLISTPRO 

MAKE IT USA · INTL WATCH LIVE





POLITICS

## Supreme Court hands victory to blind man who sued Domino's over site accessibility

PUBLISHED MON, OCT 7 2019-9:40 AM EDT | UPDATED TUE, OCT 8 2019-6:31 PM EDT




**Tucker Higgins**  
@IN/TUCKER-HIGGINS-SB162295/  
@TUCKERHIGGINS

SHARE    


KEY POINTS

- The Supreme Court denied a petition from pizza giant Domino's on Monday to hear whether its website is required to be accessible to the disabled, leaving in place a lower court decision against the company.
- The case was originally brought by a blind man named Guillermo Robles, who sued the pizza chain after he was unable to order food on Domino's website and mobile app despite using screen-reading software.
- The decision not to grant the case is a loss for the company and a win for disability advocates, who have argued that if businesses do not have to maintain accessible sites, disabled people could be effectively shut out of substantial portions of the economy.


TRENDING NOW



1 Wells Fargo tells customers it's shuttering all personal lines of credit



2 What the impending rubber 'apocalypse' means for the U.S. economy



3 This 40-year-old is paying off his car loan early—here's why that's a smart approach

# Why?

## It improves SEO and UX

---

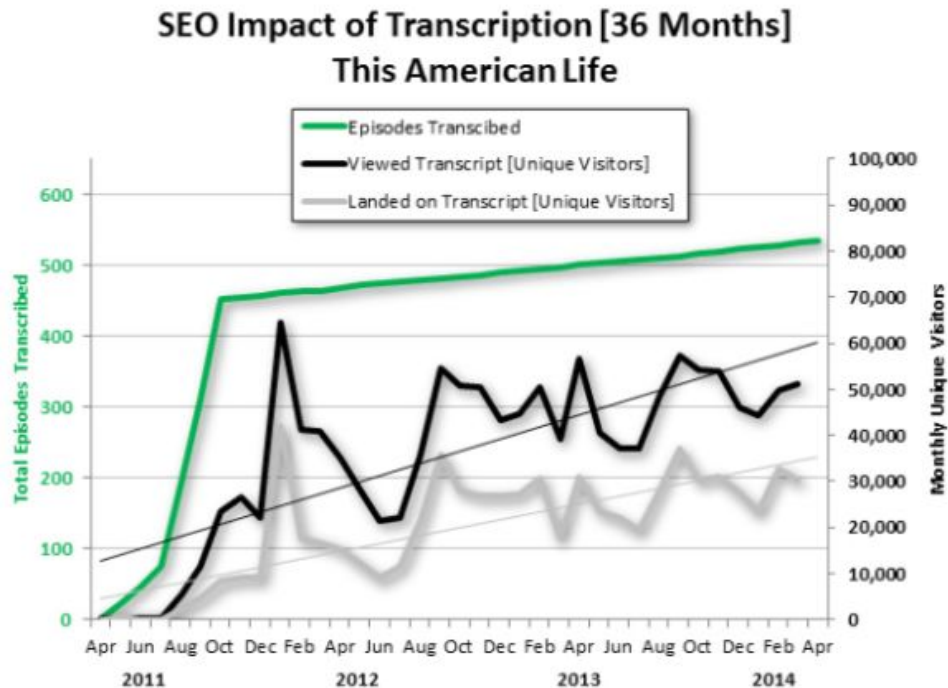
A lot of the work needed for improving a11y actually improves SEO and UX.

Examples: Site maps, video and audio transcriptions, those generated Yelp pages

# Why?

## It improves SEO and UX

Transcribing “This American Life” podcasts improved SEO, and also increased the number of ESL visitors because it was easier to translate.



Source: 3PlayMedia

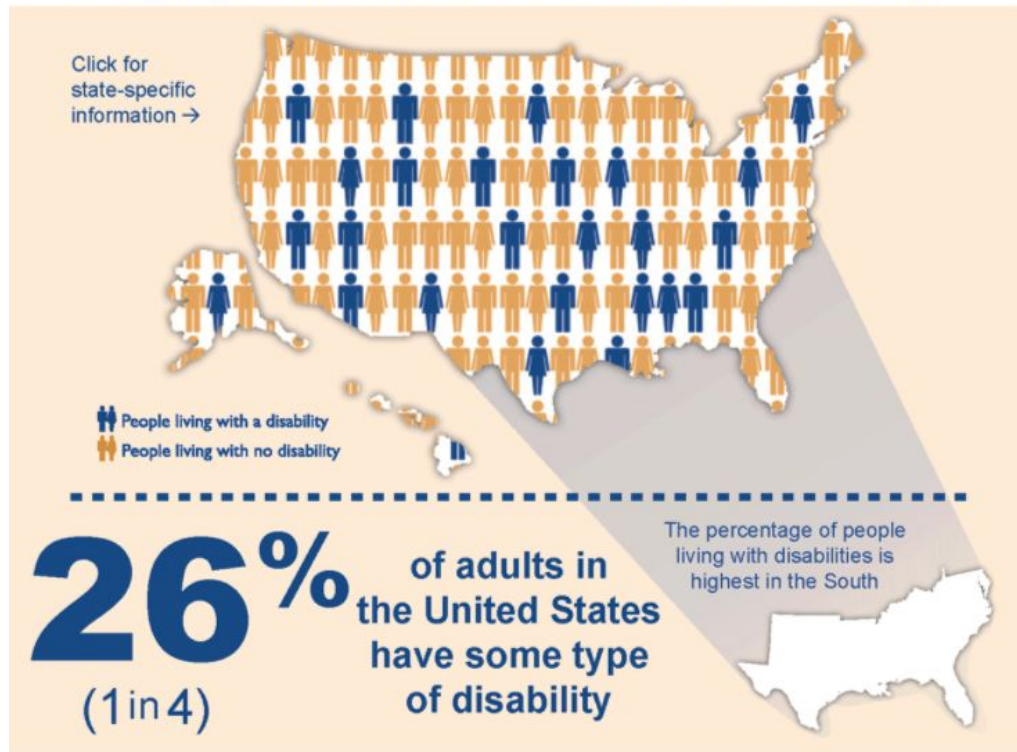
# Why?

## It expands your user base.

The data shows that the global market of people with disabilities is over 1 billion people with a spending power of more than \$6 trillion.

Source: The CDC

**61 million** adults in the United States live with a disability





# Why?

It expands your user base.

Don't be like Beyoncé!  
Invest in users early!



**It's the right thing to do!**

# More about folks with disabilities

# Categories of disability types

## **Visual**

Blindness, color blindness, low vision, glaucoma

## **Hearing**

Deafness, hard-of-hearing

## **Motor**

Limited fine-motor control

## **Cognitive**

Difficult focusing, learning disabilities

## **Seizures**

# Different ways of interacting with a product

- **Screen Readers**
  - Assistive technology that reads what's on the screen, most commonly used by those with blindness or low vision.
  - Try out [VoiceOver](#) if you have OSX. Start it with Command + F5
- **Keyboard**
  - Some folks do not use a mouse and only use the keyboard
- **Browser Zoom** (ideally support up to 200%)
- [Custom styling](#)
- **Other assistive technology**
  - Refreshable braille displays
  - Magnification software

**Let's develop some  
empathy**

# How would you feel if you experienced a website like this?

[Bad navigation example](#)

[Good navigation example](#)

How do we evaluate  
accessibility?

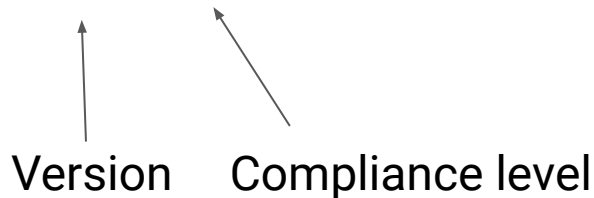


# Web Content Accessibility Guidelines

**WCAG** is a set of requirements published by W3C.

You might see

“We need to be **WCAG 2.1 AA** compliant”



# WCAG Compliance is like HIPAA Compliance

A team can't "solve" accessibility at a company; rather, they must make a commitment to continually address it.

Just like how HIPAA compliance is an ever-changing standard as the healthcare industry and the internet expands, WCAG compliance, too, is always evolving and isn't confined to a small set-and-forget list of TODOs.

# WCAG Compliance

For example, [for time-based media](#), here are some examples of the differences between compliance levels:

- A — “Minimal Compliance” — Captions for **pre-recorded** video/audio, alternatives provided for pre-recorded video/audio (e.g. transcriptions)
- AA — “Acceptable Compliance” — Everything in A *and* captions for **live** video/audio, additional narrations provided during pre-recorded video
- AAA — “Optimal Compliance” — Everything in AA *and* sign language interpretation for video, more detailed narrations provided during and in pauses for pre-recorded video, live transcriptions

# WCAG Compliance

[Compliance documents](#) exist for each version

But no matter which version, the four main guiding principles of WCAG are:  
**perceivable, operable, understandable, and robust (POUR)**

**If your eyes just glazed over,  
that's ok.**

Just remember that you want  
WCAG 2.1 AA Compliance

Let's go back to that  
POUR thing, shall we?

# P. O. U. R.

**Perceivable** - information needs to be available to at least one sense (sight, hearing, touch)

**Operable** - a user must be able to perform all interface actions (i.e., with a keyboard)

**Understandable** - a user should understand the language on a page and how to operate it

**Robust** - different user agents, including assistive technologies, content should be able to access all content



# Some questions to ask while auditing accessibility

- Can I use the keyboard to do everything a mouse can do?
- Is the screen reader giving me all relevant info? (Role, content/name, state)
- What happens when we set the browser zoom to 200%?

# Testing tools

# WAVE browser extension/website

## [WAVE browser plugin](#)

- This extension will audit your page for you with helpful error and warning messages
- Remember: automated audits can only find ~30% of accessibility issues. The work isn't done when WAVE returns 0 errors.

# Allly Linters & SDE Plugins

There are now linters and plugins that help catch and automate accessibility fixes as part of your normal linting process

- There are [Eslint linters for React projects](#)
- [React Axe](#) spits a11y errors it finds into the dev console while you're running in development

But note, like WAVE, these catch only ~30% of accessibility issues. Certain issues, like an element not having a focus state, might not be found.

# Screen readers

If you have time, do a quick pass through with a screen reader and your keyboard before opening a PR.

Never used a screen reader? Practice on <https://screenreader-practice.glitch.me/#>

## Screen readers

- VoiceOver on OSX
- JAWS on Windows
- Linux Screen Reader (LSR) on Linux
- In-browser [ChromeVox](#) on Google Chrome

So...How can I start  
prioritizing accessibility?

**Great question.**

**Great question.  
There's no silver bullet way.**



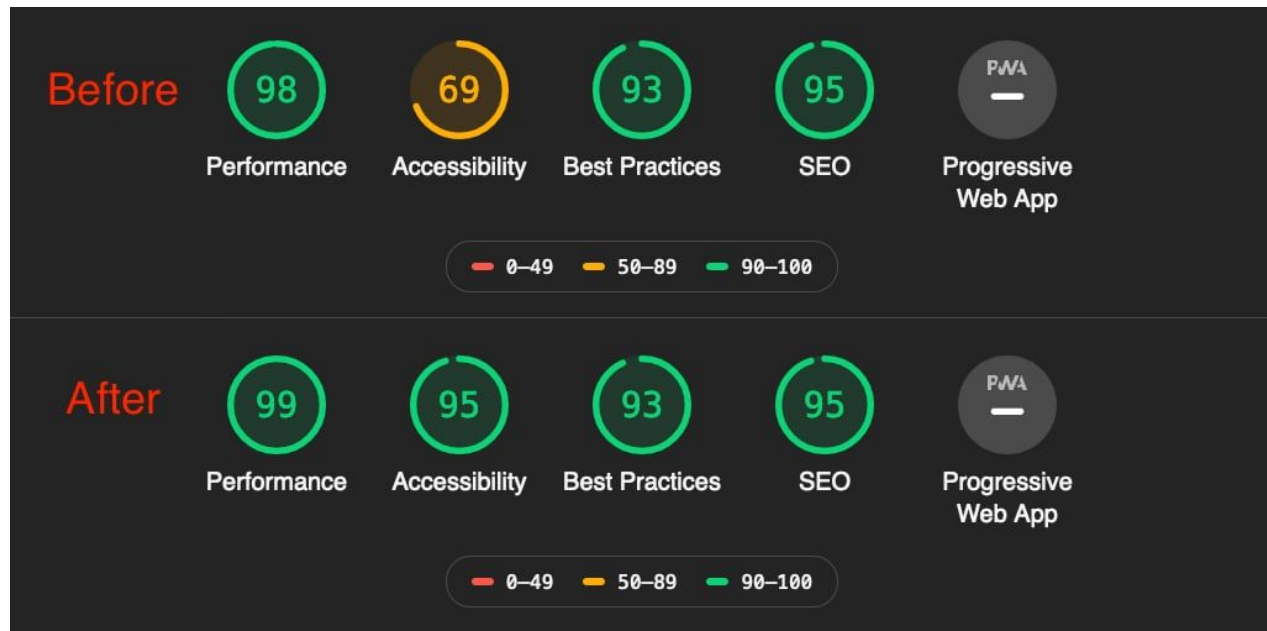
**But here's some ideas.**

# Create milestones

Treat accessibility like any other measurable metric, like a test coverage or performance goal.

## Set a quantifiable goal

You could select a specific WCAG compliance as a goal, or a [Lighthouse accessibility score](#)

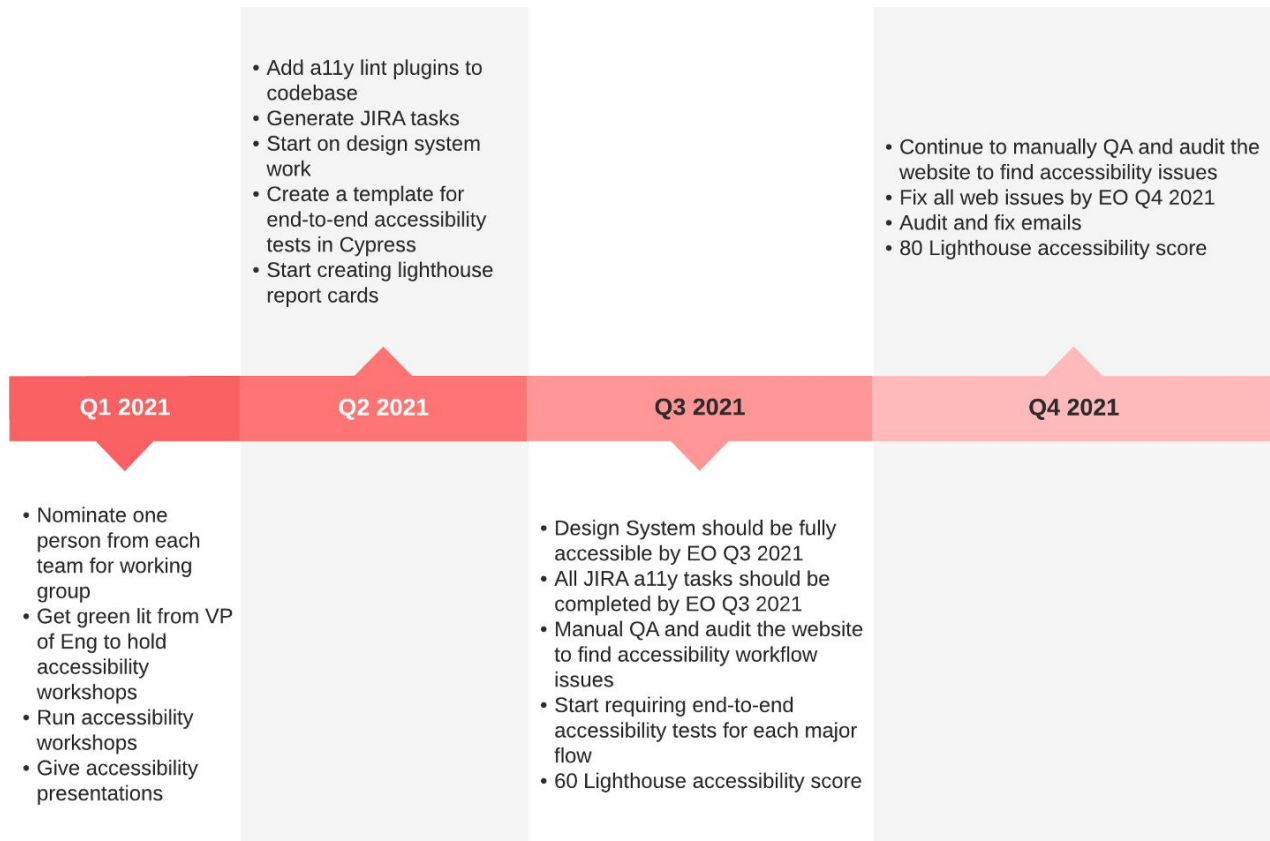


# Create milestones

We had to make a roadmap as part of one of our settlements...it actually really helped whip us into shape.

It could be a good idea to do this **BEFORE** you get slammed with a lawsuit!

## Make an ally roadmap



# Audit everything

A11y issues are bugs.

That's it. End of story.

## Add a11y linters to the codebase

---

Linters only catch 20-25% of all accessibility issues, but they're still a start!

- Add an a11y linter to your codebase
- Run it initially and create TODO tasks from every single issue raised
- Prioritize a couple tasks every week
- Enable the linter in pre-commit hooks such that no one can create new code that creates a11y issues



# Audit everything

You will have to manually audit. There's no avoiding it!

## Manually audit

---

Look at every big flow in your website and ensure it's keyboard accessible.

- You don't have to know how to use a screen reader to make your website accessible, just learn how to use keyboard-only
- Essential flows, especially ones that involve paying or receiving money or opting in or out of emails, should be completely keyboard-accessible
- Modals, or image lightboxes or carousels, should be usable via keyboard

# Learn together

Create a culture of learning  
and an inclusive  
environment for people to  
learn together.

## Introducing the ally working group

---

Create a group that

- Researches together
- Is highly democratic and welcoming
- Holds office hours and is available
- Has dedicated time every week to do this important work



D.igital  
A.ccessibility  
W.orking  
G roup

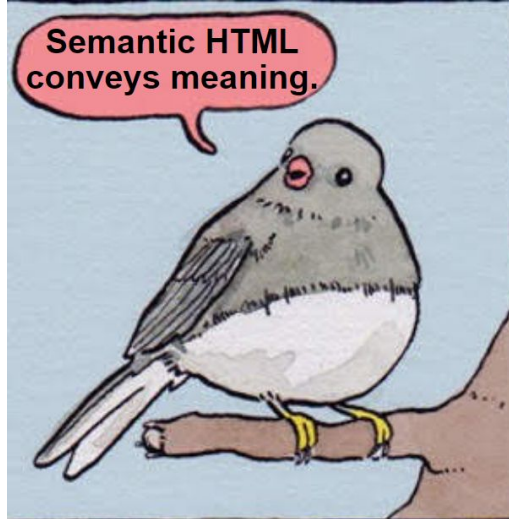
**I know what you're thinking.**



But like...how do I  
actually fix things?

**Here's a novel concept.**

Use HTML



# Use HTML

(correctly)



Standard HTML elements come with a lot of great built-in functionality.

Default to these whenever possible.

# Using HTML properly



Buy Now >

Let's say we inspect this custom button and find that it's a `<div />` with an `onclick` and some fun CSS. To make it accessible, we'd need to add:

- `tabindex=0` to add it to the tab order.
- A key event handler so that "Enter" does the same thing as `onclick`
- `role="button"` so that the screen reader knows it's a button
- Additional styling for the focus state so the keyboard users can keep track of their location on the page

Now if we just change the `<div />` to a `<button />` we'd get all of that for free!

# Semantic HTML features

- Standard HTML elements come with a lot of great built-in functionality. Default to these whenever possible.
  - role
  - Keyboard support
  - Visual focus indicators
- Use [aria-props](#)
- The element order in the DOM should make sense. Don't rely on CSS to order pieces of your page.
- Other noteworthy props are `label[for]` and `tabindex`

# Semantic HTML features (cont'd)

- Add alt text to images
  - This attribute is required
  - If the image is purely decorative it should have an alt=""
- Headers h1 to h6 should be in order on the page.  
e.g. h3 should only be after an h2.
- Be mindful of nesting.  
e.g. a <button /> should not be inside another <button />



Ally seems pretty cool.  
Where can I learn more?

# Resources

## Informational

---

- [WebAIM](#)  
A great source of a11y information
- [WAI-ARIA Authoring Practices 1.1](#)  
Everything you need to know to make common widgets and UI patterns accessible.
- [Web Content Accessibility Guidelines 2.1](#)  
The standards themselves. Dense, but useful.
- [A11ycasts](#)  
Videos by the Google Chrome Dev team
- [The A11y Project](#)  
Education around making accessible designs
- [Mozilla's a11y documentation](#)  
Easy-to-comprehend a11y documentation

# Resources

Funny thing is I don't actually offer accessibility consulting at this time.

However, there are many amazing consulting groups that do! Here are some of my favorites.

## Consulting + Auditing services

---

- [Deque Systems](#)  
A highly esteemed group that makes lots of testing and auditing tools and even offers consulting services
- [Essential Accessibility](#)  
A consulting group that offers everything from auditing, legal advice, and even technical advice
- [AccessiBe](#)  
A company that offers services in audio/video transcription tools, accessible downloadable files, and individualized consulting.
- [Tenably](#)  
A AI consultant powered by the magic of GPT-3 that can walk through your site and offer you tips in plain English.

# Resources

## Tooling

---

- [Screen Reader practice](#)
- [ANDI](#)  
An automated accessibility testing tool to run on your webpages
- [WebAIM's Color Contrast checker](#)  
A great tool to see if your text is readable
- [Deque Systems](#)  
A highly esteemed group that makes lots of testing and auditing tools and even offers consulting services
- [Lighthouse](#)  
See performance, SEO, a11y, and other scores on the same dashboard
- [AccessMonitor](#)  
A plugin that tests a11y on your WordPress site

# Resources

If you would like to read the content you were just presented in far more detail, check out these 4 articles this presentation was based on!

Also, a totally shameless plug.

## Karina's ally articles

---

- [Web Accessibility for Newbies](#)
- [How to Convince Company Leadership to Care About Accessibility](#)
- [5 Ideas on How to Incorporate Web Accessibility into your Dev Workflow](#)
- [The Developer's Guide to Web Accessibility Auditing and Tooling](#)

# Resources

If you would like to read the content you were just presented in far more detail, check out these 4 articles this presentation was based on!

Also, a totally shameless plug.

## Karina's ally articles

---

- [Web Accessibility for Newbies](#)
- [How to Convince Company Leadership to Care About Accessibility](#)
- [5 Ideas on How to Incorporate Web Accessibility into your Dev Workflow](#)
- [The Developer's Guide to Web Accessibility Auditing and Tooling](#)



## My Contact Info

@karomancer

[Twitter](#), [Medium](#), [LinkedIn](#), [Dribbble](#)

karinachowtime@gmail.com

<https://www.karinachowtime.com>