



I wanted to come here today and tell you a story.

I wanted to try to make accessibility personal, something everyone here could relate to.

So I sat down to write a story.

Except, no one in my immediate family is differently-abled in a way that affects their relationship with technology.

My father isn't blind; I don't have a color-blind brother or a grandmother with arthritis. I've never had to help my mom navigate a website.

I don't care about accessibility because someone in my life needs it in order to use technology. So I realized the way to make this personal, is to tell you why it's personal *to me*.



I care about accessibility because really enjoy making quality experiences....and because it's my job.

I work for JP Morgan Chase, where we use Ember to build ambitious things.

My team provides accessible UI components through an Ember addon that other teams use to build corporate and investment banking applications.

We have a federal requirement to provide accessible products.

When I joined the team, we didn't have very good guidance about how to make things accessible. So guess what? I learned it!



I care about accessibility because I care about web standards.

Over the last 22 years, I've discovered that the more I get to know the specs, the more clearly I can think about how to architect a solution.

I can build new things because I understand the context of how things are *already* supposed to work.

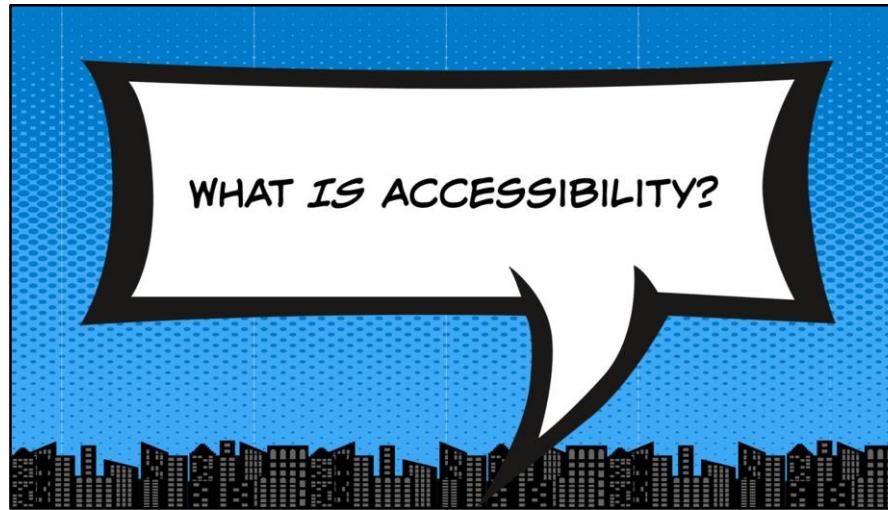


There are also accessibility spec - known as WAI-ARIA
Web accessibility initiative for Accessible rich internet applications
Phew that was a mouthful!

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

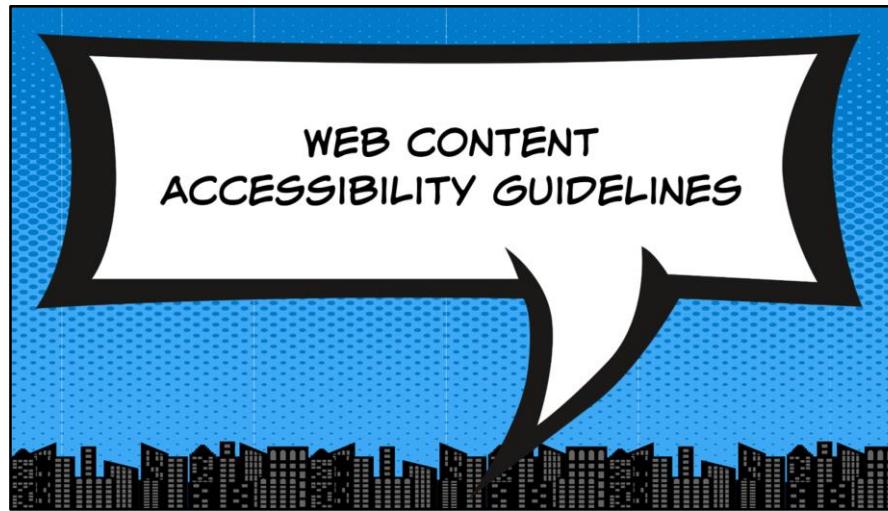


That reminds me! Before we continue, let's look at some other lingo you might commonly hear in the accessibility space.

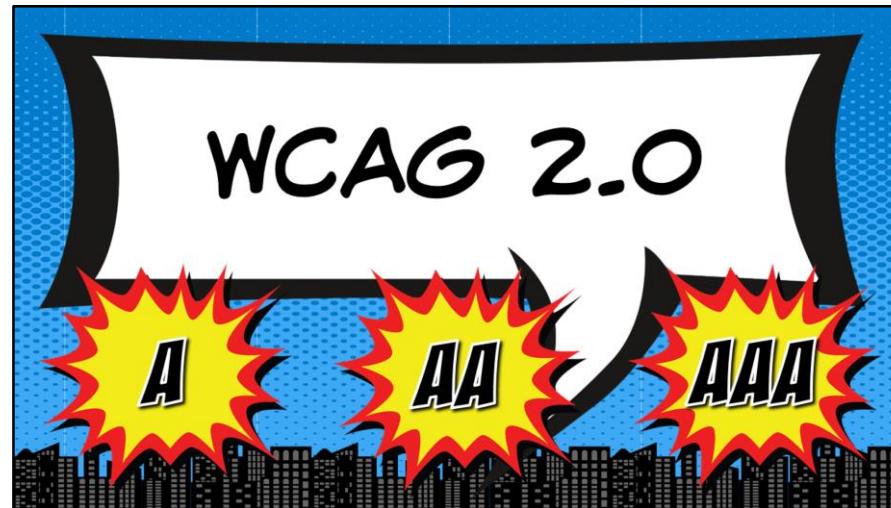


Accessibility is the practice of making your websites usable by as many people as possible

Can users of all ability levels use my application in the way it was designed and intended to be used?



Another common term is Web Content Accessibility Guidelines, or



WCAG, for short

WCAG is currently on recommendation version 2.0.

There are three levels-

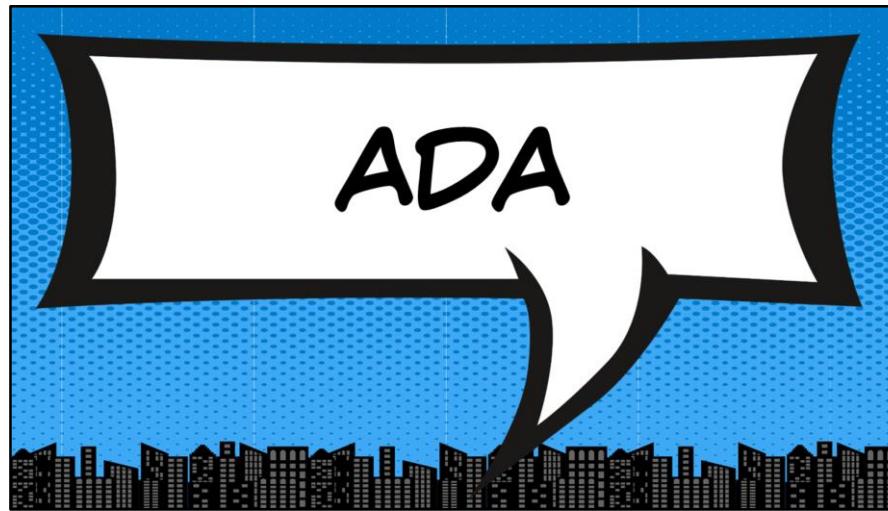
A is the first level,

AA includes A and extends it – and is the most common for companies with a legal requirement to provide conformance

and AAA, which includes and extends both.

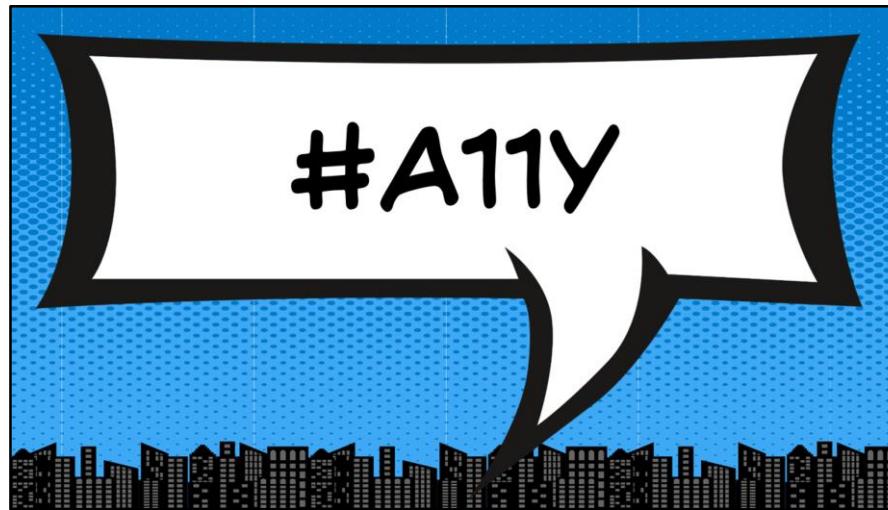


The Americans with Disabilities act is often referred to as



ADA

Other countries like Australia, Portugal, Canada and the UK have similar laws.



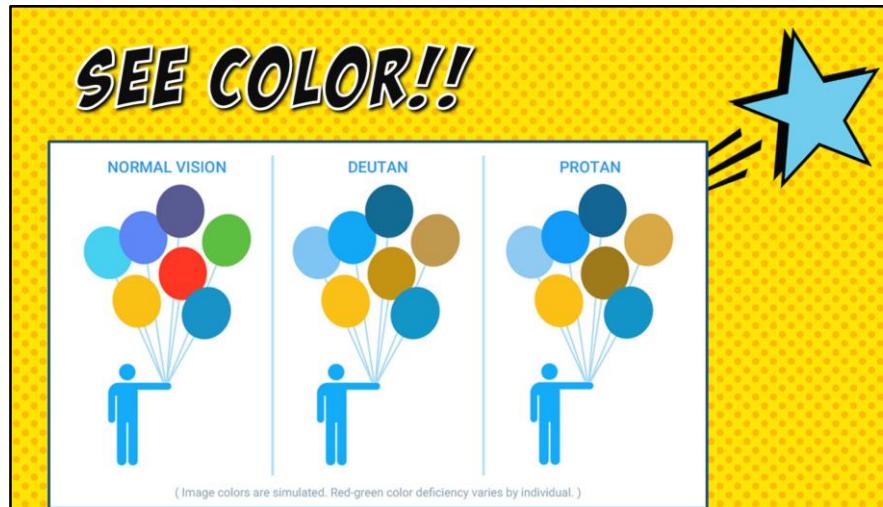
If you're one of my twitter followers, you will see me use this hashtag a lot!

If you look at the word 'accessibility' you will see that there are 11 letters between the 'A' and the 'y' - hence A-eleven-Y.



I care about accessibility because the future is now!!

There have been a lot of really great advancements in assistive technology in the last few years- let's look at some of them.



EnChroma glasses give color-blind people the ability to see color- did you know that there are an estimated 300 million people in the world with a color vision deficiency? Did you also know that many people with color blindness cannot tell that the power connector on a MacBook changes color? Did you know that people who experience color-blindness are often surprised to learn that peanut butter is not green?



Doctors at the UNC School of Medicine in Chapel Hill performed the first auditory brain stem implant surgery, which gave a deaf child back his hearing. This is a photo of the moment when he heard his father's voice for the first time.

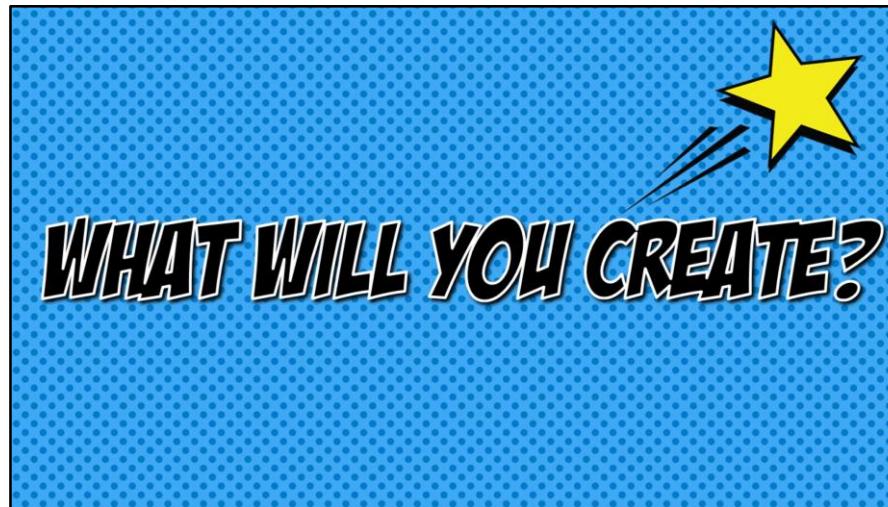


Microsoft has developed a cloud service called Video Indexer

- It can provide transcriptions, and extract text that appears in video that isn't spoken
- Live translate to different languages
- analyze sentiment based on the speaker's facial expressions
- Even adjust the language model used for transcription – making it industry or market specific. This means you could make your content more accessible to viewers who haven't had as much access to education.



Include.ai is a project started by a student at Stanford, who wanted to help his father have better access to image descriptions on the internet.

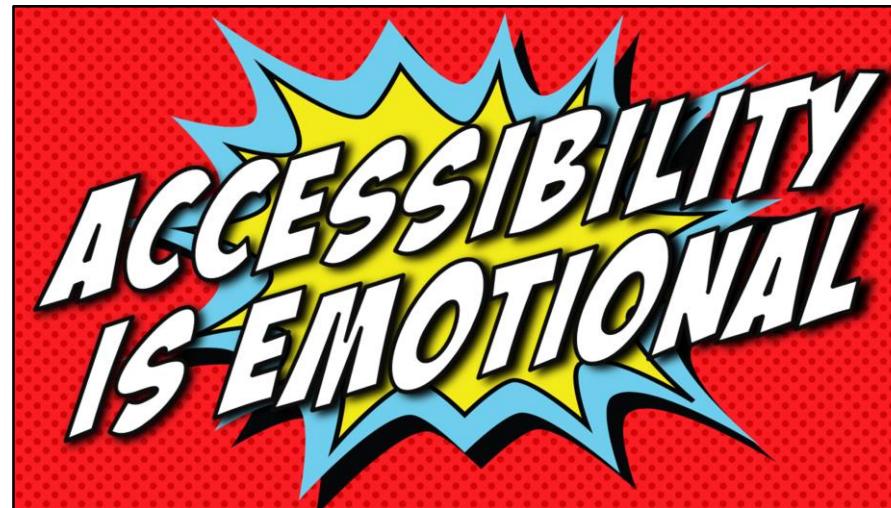


What about you? What will you create? Brilliant, creative minds are in this room, each of you with the potential to solve hard problems in ambitious ways. When I see these advances in technology, I am inspired. I feel empowered to be curious. To dream. To think outside the box.

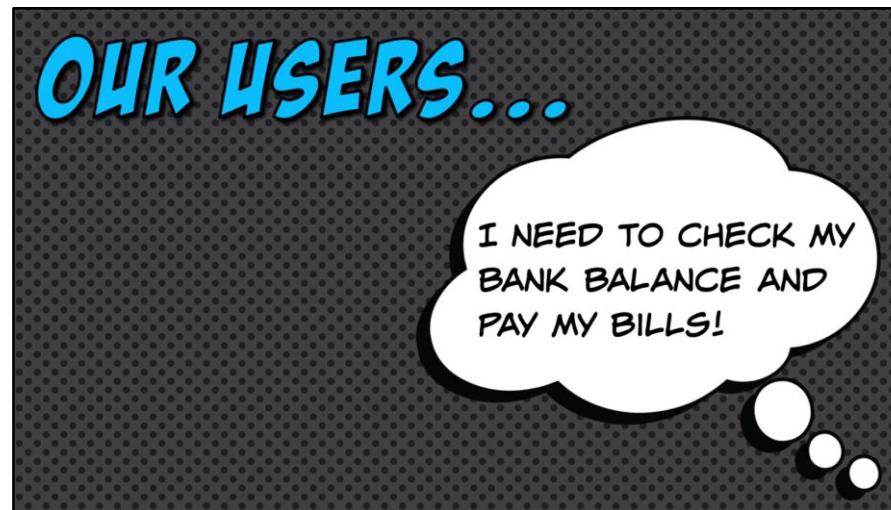


Even in our own community there is innovation in how we provide accessible solutions. I'm absolutely delighted to be able to introduce you all to the official Ember-A11y Zoey and Tomster, thanks to the talented Lindsey Wilson. To me, these represent a tangible commitment to building accessible things in Ember.

So, we'll go over accessibility in Ember and Ember addons in a minute, but I think it's important for us to first talk about something else, some of the things that are hard about accessibility.



Accessibility is emotional.



It's emotional for our users, who need to perform tasks on the web to get through their daily lives.



Who want to buy presents for their families.



Who want to use your new website...but they can't.

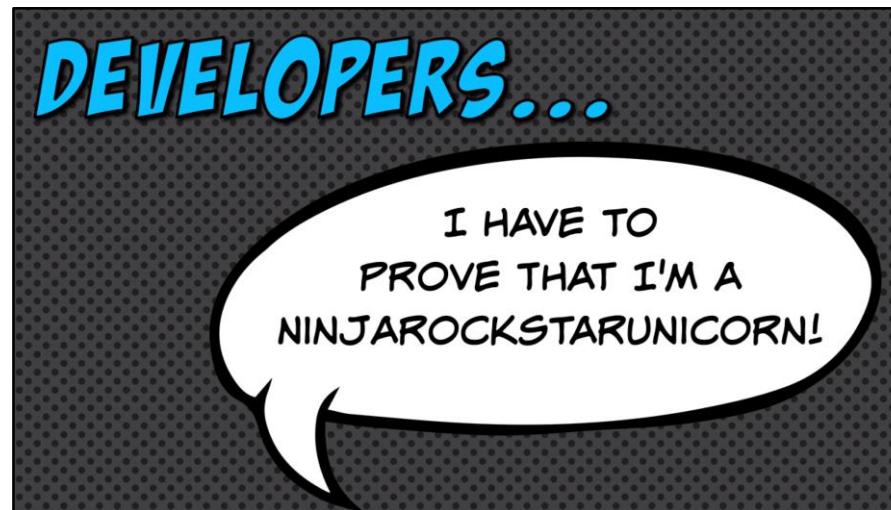


They feel frustrated and isolated and once again, wondering why programmers don't care.



Accessibility is emotional for developers, too.

- we're under pressure to finish projects that have been promised at an unrealistic timeline
- We're being told to "ship it now and fix it later" – but is there really ever a "later"? Are we failing our users when we fail to be the technical experts we were hired to be?

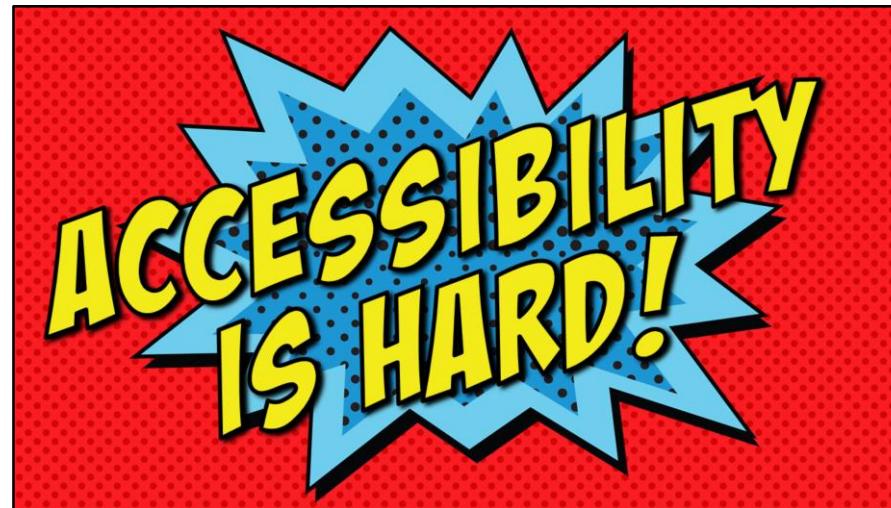


Or maybe it's that we feel pressure to prove ourselves. Has anyone here felt the pressure to prove themselves? I know I have.

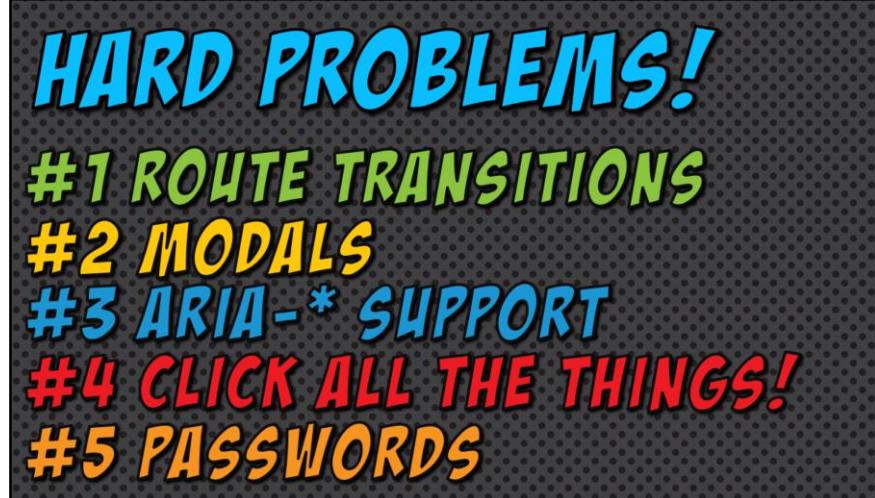
We also are under pressure to prove ourselves- some of which comes from inside our own minds.



Under the weight of all this, it's no wonder that accessibility is easy to ignore.



Accessibility can also be difficult to get correct when you're first starting out.



We will all have things that we think are the hardest to do, but this is my list of "top five hard a11y things that I'd like to see permanently solved in Ember"

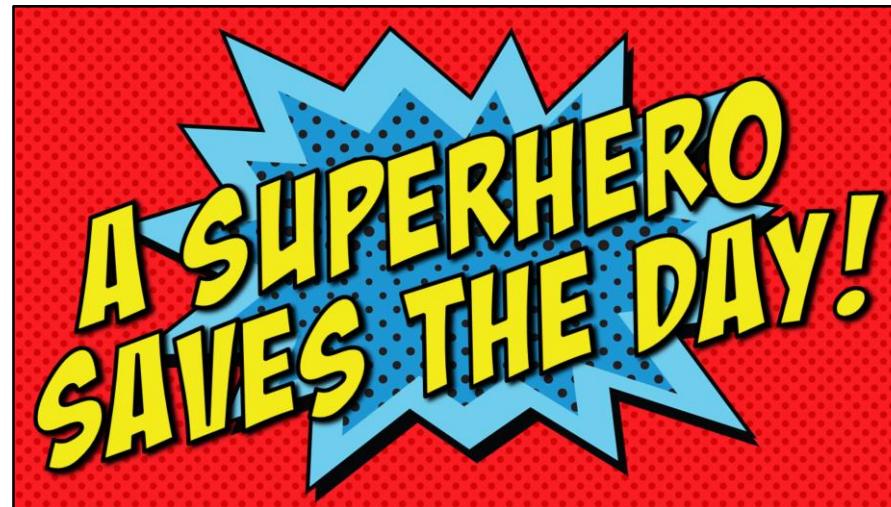
- Route transitions. I think it's incredible, the way Ember handles routing....but one of the downsides is that when the route changes, nothing is announced to the screen reader other than a link being clicked. If you have to depend on only on what the screen reader is telling you, you'd have no way to know that the link did what it was supposed to do. And- this isn't just an issue with Ember, it's an issue with all SPAs out of the box.
- Modals that insert themselves correctly into the DOM, have the correct roles, and most importantly, trap focus!

Plus, there's a difference between visible focus and screen reader focus.

- ARIA attribute support- there's a specific set of rules for what things should have labels or descriptions, and what roles should be on what element in specific context. Of course, now that I think about it, it IS a specific set of rules...and we're engineers and we like rules....hmmm....
...maybe we *could* engineer for this....
- You can add an action to anything. This is awesome. This also not awesome, because it's not just the click that you need. If you can click on it, you also need to be able to interact with it via keyboard. A screen reader needs a machine readable name for what kind of interactive element it is, because that tells the person with assistive technology what to expect, or what to *do* with it.
- An interactive element also cannot contain another interactive element.
- So this one is a bit messy, turns out. I wonder if we could improve this?
- Passwords are hard but they're not just hard in Ember- they're hard everywhere. Have you ever tried to fill in a password with a screen reader turned on, while not looking at the screen? I'll give you a hint... "star, star, star, star, star, star..." How do you ensure privacy but still provide an inclusive experience? I wonder if this is an area where we need to look differently at the problem.



That's quite a bit of work.



But, there are some things we can do. There is a superhero to save the day.



I bet you thought I was going to say that I was the hero, right?

Or maybe something cheesy like “there’s a hero in all of us”

Yeah, nope.

Web standards. Web standards are the superheroes for accessibility.

Web standards give us ways to pick up easy wins. Wins by default.

THE "EASY" WINS!!

#1 USE HTML5

#2 LINK VS. BUTTON

#3 KEYBOARD NAVIGATION

#4 COLOR CONTRAST

#5 IMAGE ALT TEXT

What are some of the relatively easier things that you can do?

- Use html5. use html5 use html5 use html5. I hope you hear this over and over and over again in your head. Any time you go to reach for a div when you should be using a native html5 element, I hope you hear me saying, "use html5"
- Use a link if it's a route. Use a button if it does a button thing like submit a form.
- If you can do something with your mouse, you have to be able to do it with your keyboard. Add keyboard navigation to your apps!

- Check your color contrast. You don't even have to write your own, there are like a thousand out there.
- Add alt text to your image, if it's relevant to the subject.



Here's what I mean. Is it just for show? Maybe it's breaking up a giant wall of text? Then, you'll want to leave the alt tag blank and add the role presentation.

BUT

ALT TEXT TLDR

```

```



If the image is what you need in order to figure out if that really is the toy your nephew asked for, you'll be wanting something that is descriptive. Note that I don't just say "Elmo toy" here, I'm trying to use words to describe what it really looks like.



Of course, despite these relatively easy wins being available, there will still be developer fights—some of these are, to me, incredibly irrelevant to the product I'm actually trying to build.

Like tabs vs spaces.



I mean, you can fight about it if you want. I never *really* cared about this particular fight. Sorta like Batman vs. Superman. I knew the fight was going on, I knew they were superheroes and it would therefore be epic, but in the end, I didn't really care which one of them won.

I care about accessibility so I'm not going to tell you to not fight, I'm going to propose what I think is a more interesting thing to fight about.



How about your position on the DIV element instead of a native semantic html element?

Sure, adding a role of heading and an aria-level to your div is *technically* fine,



But this is what happens if you use semantic html.



So which one is easier? Which one is really “fine”?



Now, if I'm going to have to get behind a fight that isn't Black Panther vs Gold Jaguar,
then a fight about semantic html vs not semantic html is probably the one I'd prefer.

I mean, if you have to fight. Maybe the kind of fight where it's like Hulk and Thor on
Ragnarok.

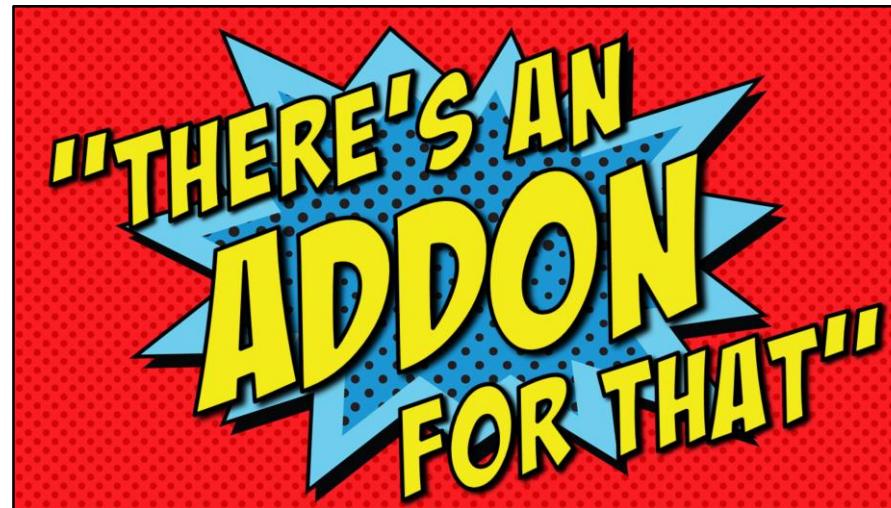
An epic-but-we're-still-friends-after kind of fight.



I wonder if this is perhaps a place we could all improve.

or, alternatively, more permanently solve in a “this is the modern web and our div approach isn’t going anywhere.”

Either way- it would be beneficial for us to resolve the issue, because we are excluding users.



Okay so, let's get back to Ember.

THESE ALREADY EXIST

EMBER-A11Y
EMBER-A11Y-TESTING
EMBER-A11Y-LANDMARKS
A11Y-ANNOUNCER
EMBER-CLI-TEMPLATE-LINT
EMBER-COMPONENT-ATTRIBUTES

I was working on a huge Ember addon, and I needed it to be accessible. There are *a lot* of addons out there, but too many of them have accessibility issues.

So I got involved! I've found some accessibility-related addons, and even wrote one of my own!

Well, there are *some* addons for that....but not nearly enough. We have more problems to solve.

KEEP IN MIND...

- #1 KEYBOARD NAVIGATION**
- #2 FOCUS MANAGEMENT**
- #3 ARIA-* SUPPORT**
- #4 RENDER SEMANTIC HTML**
- #5 REMEMBER!**

If you're an addon contributor, here are some things to keep in mind.
I wonder if you could make these a criteria for your addon?
My last protip is – remember.



YOU'RE NOT ALONE!

Remember that you're not alone.
You might not know how to do this.
Hey guess what?
You don't have to pretend that you do.
You just don't know how to do it YET.
So remind yourself, you're not alone. This isn't just your responsibility. You CAN learn how to make accessible Ember addons and applications.
And when you get stuck? Just ask. I still have to ask! It's not a big deal to ask.
Come to the a11y channel in the Ember Community Slack Chat! We're here to figure it out together!



Whether you have it all figured out, or you don't.... Be magnanimous.
I love this word. This word means we're gracious, but we still have our confidence. We
still have that little bit of ego.
(Kind of like Aquaman.)
Let's be magnanimous. Let's rise above. Let's think bigger.

MAGNANIMOUSLY SOLVE

#1 ROUTE TRANSITIONS

#2 MODALS

#3 ARIA-* SUPPORT

#4 CLICK ALL THE THINGS!

#5 PASSWORDS

What if the solutions for the hard problems were in Ember...by default?

What if we became the first framework to be like "yo, we got this."

What if we pioneered yet another hard problem for the entire JS Framework community?

Because, these are the kind of solutions the world needs- not yet another build tool.



Get involved.

My friend Jetta says, "if you want to show up big, you have to be big" and this is how I think about Ember. Ember shows up big and does ambitious things.

And, that's really the crux of my story- I got involved.

My future story is that I'll stay involved. I'll keep inviting you to be involved.

We CAN do this.



You can find me as melsumner on Ember slack
Or melaniersumner on twitter
And if you're ever in Chicago, we'd love to see you.

Thank you!