

# DESIGNING ACCESSIBLE COMPONENTS



image source: [Heydon Pickering](#)

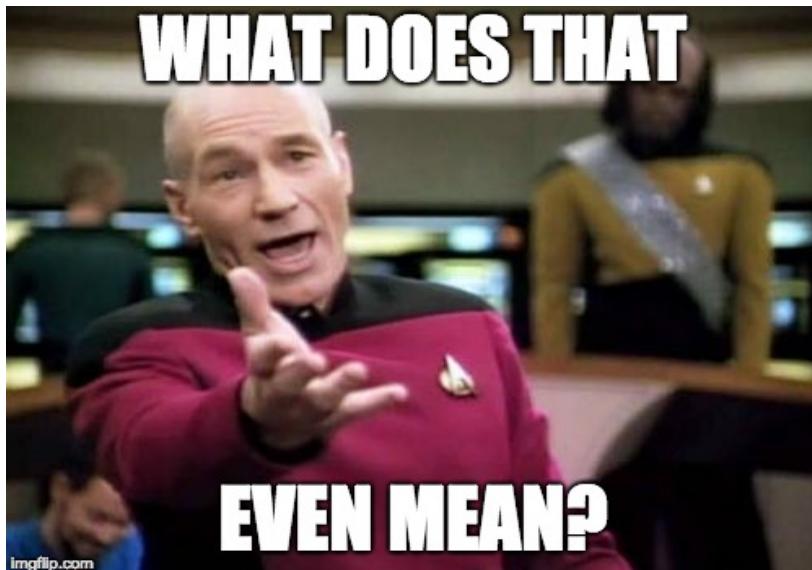
# Who is this Sarah person?



- Dojo widgets
- web accessibility
- herds cats



# DESIGNING ACCESSIBILITY? OTHER DEVELOPERS? WHAT?



# TODAY WE'RE TALKING ABOUT:

- Component libraries
- Diverse models of user interaction
- Diverse models of developer meddling

# OK BUT... WHY DO I CARE?



The Rolling Quads. Source: [99 Percent Invisible](#)

# ACCESSIBILITY IS:

- Freedom
- Privacy
- Participation
- Equality

# THE CURB CUT EFFECT



Necessary

(source: [the Smithsonian](#))



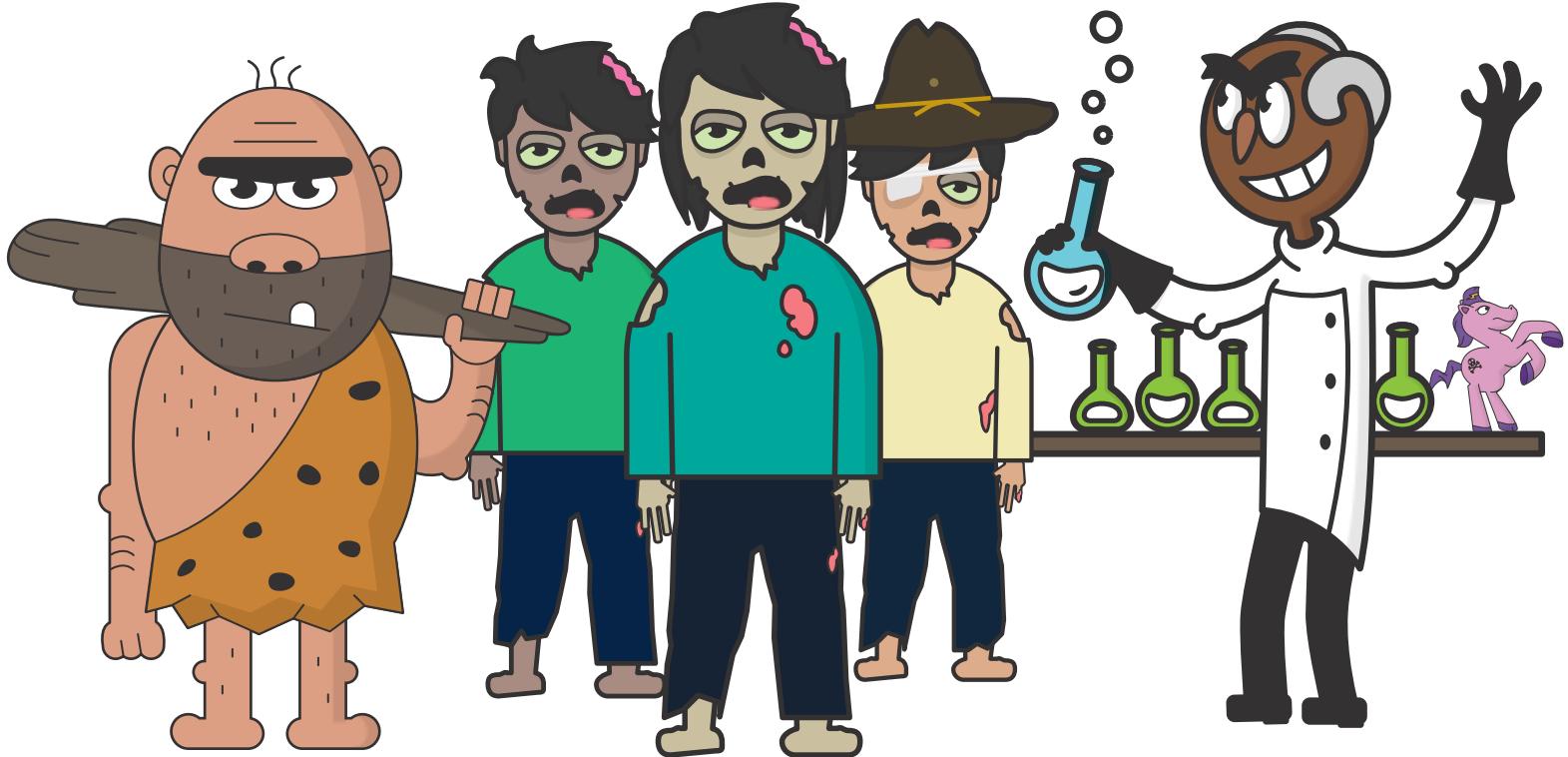
Beneficial

([source](#))

- Curb cuts
- Audio captions and transcripts
- High contrast mode or night mode
- Distraction-free features
- An accessible, semantic web?

# OK, ACCESSIBILITY. BUT WHY MAKE A PATTERN LIBRARY?

# MEET YOUR COWORKERS





*“If it looks right, I'm done.”*



*“What's an ARIA? I don't like opera.”*



*“I need these twenty obscure features, and a pony.”*



# FLEXIBILITY AND CONTROL



# AN ICON BUTTON CASE STUDY:

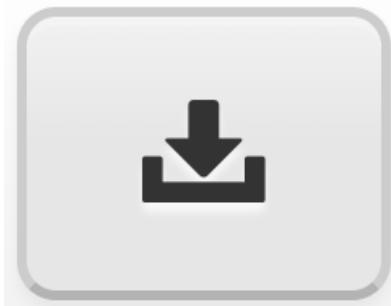
I'm a button!

Icon Button Q



# AN ICON BUTTON IN THE WILD

```
<button ng-click="triggerDownload()" class="btn"  
    tooltip="Download as a zip file">  
    <i class="icon-download-alt"></i>  
</button>
```



VoiceOver output: "button"

```
<button type="button">I'm a button!</button>
```

```
<button type="submit">  
  Search  
  <svg role="img" aria-hidden="true">  
    <use xlink:href="#search"></use>  
  </svg>  
</button>
```

```
<button type="submit">  
  <svg role="img" aria-labelledby="search-title">  
    <title id="search-title">Search</title>  
    <use xlink:href="#search"></use>  
  </svg>  
</button>
```

## OTHER POSSIBILITIES:

- aria-label
- icon fonts
- a text node with some fancy .sr-only CSS

NONE OF THIS IS EASY!

# RESERVING CONTROL:

```
interface ButtonProperties {
  type: 'submit' | 'reset' | 'button';
}

interface IconProperties {
  use: string;
  label?: string;
}
```

# FLEXIBILITY





## Dance with ponies



What could go wrong?

# COMMON BUTTON USE CASES:

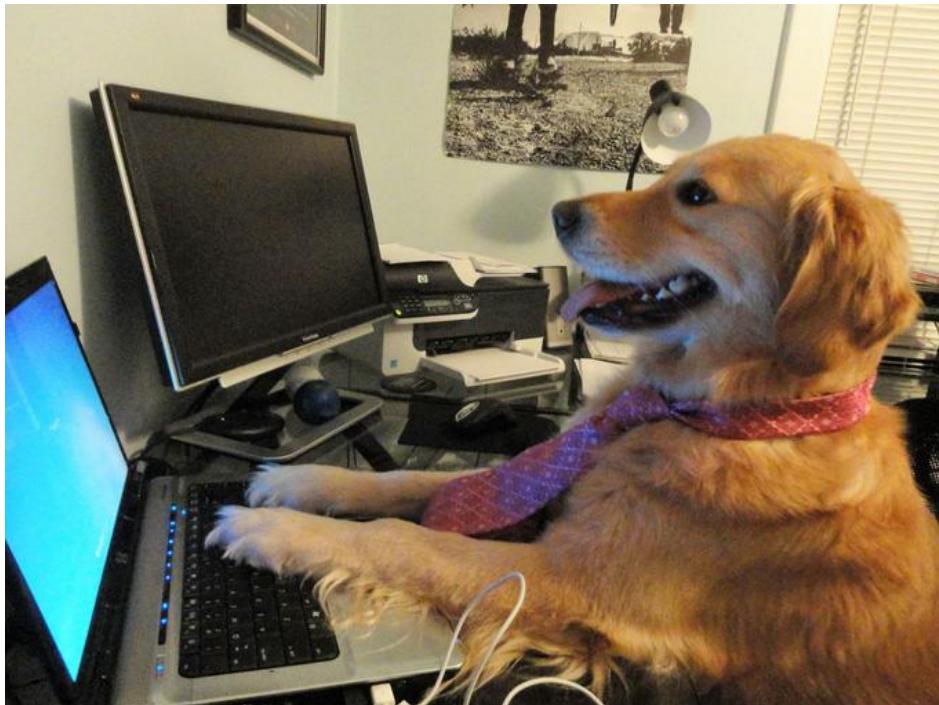
- Submit or reset a form `type="submit" | reset"`
- Open a panel `aria-expanded="true"`
- Open a menu `aria-haspopup="menu"`
- Toggle a state `aria-pressed="true"`

# UNCOMMON BUTTON USE CASES



???

# WHO REALLY KNOWS BEST?



Definitely this guy

With too many granular details left open-ended...

... you end up with this:

```
<button aria-haspopup="true" class="btn btn-popup">  
  Tooltip  
  <svg role="img">  
    <use xlink:href="#plus"></use>  
  </svg>  
</button>
```

or this:

```
<button aria-pressed="true" aria-expanded="true"
  class="btn btn-open btn-pressed">
  Open Accordion
  <svg role="img" aria-labelledby="icon-title">
    <title id="icon-title">Expand accordion</title>
    <use xlink:href="#plus"></use>
  </svg>
</button>
```

But without enough flexibility, your component might be abandoned in favor of this:

```
<div role="button" onclick="ponyDance()>  
  Ponies for all  
    
</div>
```

## BUILD EXTENSIBILITY INTO:

- Visual styles
- Layout
- Child content

## PROVIDE AN INTUITIVE API FOR BUILT-IN DEFAULTS:

```
interface ButtonProperties {  
  type: 'submit' | 'reset' | 'button' | 'toggle'  
    | 'menu' | 'accordion';  
  disabled: boolean;  
  pressed: boolean;  
  open: boolean;  
}
```

# WHERE SHOULD YOU KEEP CONTROL?

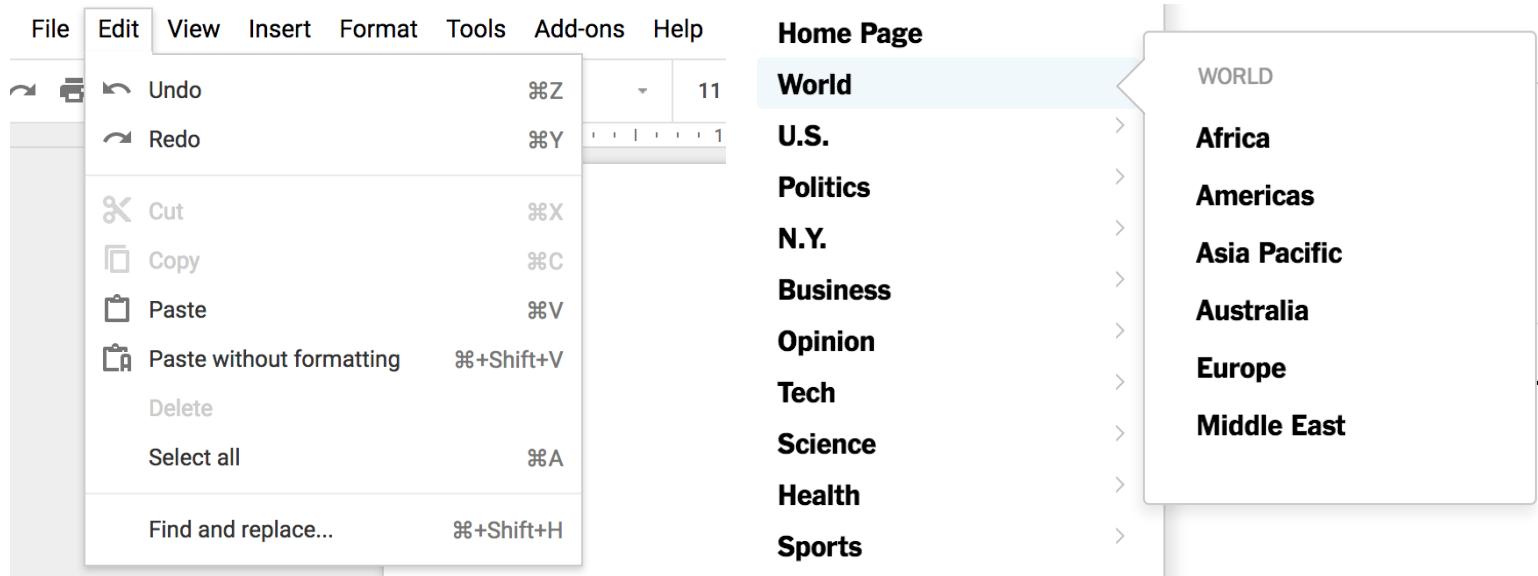
- Accessibility
- Internationalization
- Performance
- ...

# CHOOSING WHAT TO BUILD



Not everything is what it seems

# ARE THESE BOTH MENUS?



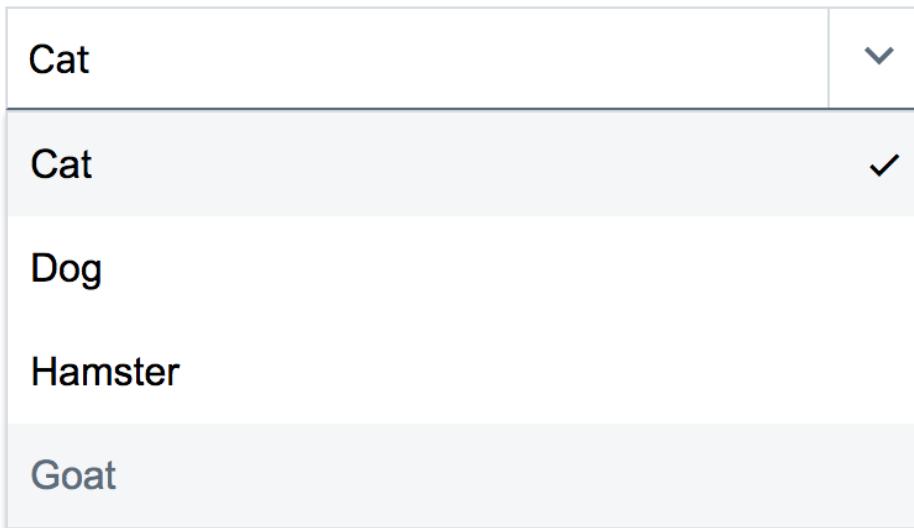
Don't ask

What it looks like

ask

What it does

# TRY DESCRIBING YOUR COMPONENT'S FUNCTION IN A SINGLE SENTENCE:



"A combobox is a normal text input enhanced with a list of suggested options."

Extra Small

Small

Medium

Large

Extra Large

✓ Alice

Bob

Charlie

Danielle

Add to calendar

Bookmark

Set alarm

Get directions

"Chips allow users to enter information, make selections, filter content, or trigger actions."

# THE LIMITS OF A FRAMEWORK



You will never be able to:

- Write appropriate labels and alt text
- Take care of overall page structure (landmarks, heading hierarchy, focus order)
- Account for all possible use cases

Instead:

- Be realistic up front
- Provide good usage documentation
- Educate other developers about diverse interaction patterns

# THANK YOU!