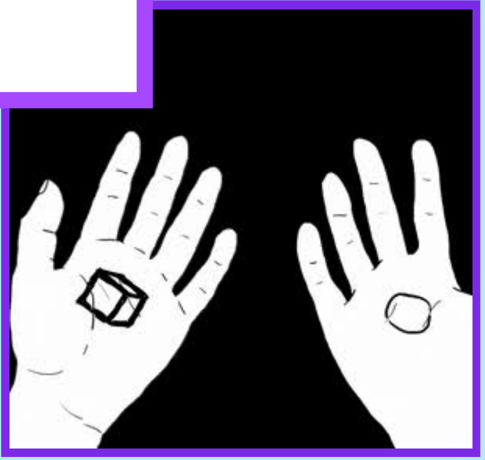
Accessibility Part 4

Presented by
Mike Nam-Lee



4. Robust

Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.

Parseable HTML

The HTML must be parseable. A lot of this comes for free as part of modern JS frameworks so I won't spend too long on this one.

Opening and closing tags and attributes must not be malformed.

```
<!-- Invalid -->
<p>Open tags must have an equivalent
close tag

<!-- Valid -->
<p>Open tags must have an equivalent
close tag</p>

<!-- Invalid -->
<input value="Self-closing tags must
have a trailing slash">

<!-- Invalid -->
<input value="Self-closing tags must
have a trailing slash"/>
```

Parseable HTML

Checks that won't be caught by modern JS frameworks:

- A doctype is defined
- No duplicate IDs
- Referenced IDs in aria-labelledby or htmlFor exist
- Only relevant attributes are added to tags

```
<!-- All HTML pages must start with -->
<!DOCTYPE html>

<!-- Be careful of not defining multiple
IDs -->
function Component() {
  return <div id="componentId"/>;
}

function Component2() {
  return <><Component/><Component/></>;
}

<!-- Instead, add some prefix to
guarantee uniqueness -->
```

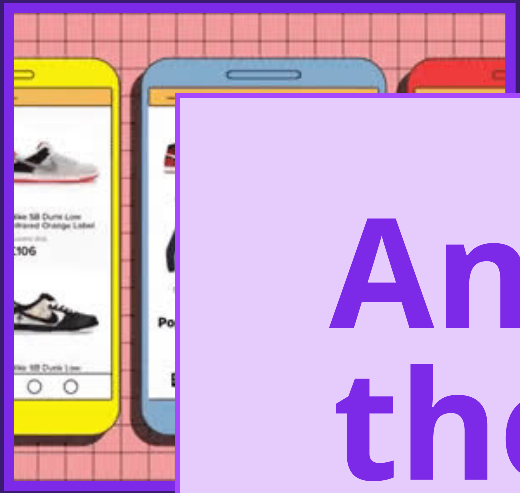
Elements with Roles

Interactive elements must either have a role or use a special tag that implies a role. As long as you're using buttons, inputs, and links, this should not be a problem. If you define your own custom interactive elements (e.g. anything with an onMouseDown, onMouseUp, onClick), you must add a role.

Any roles that you do add must be appropriate for the UX.

```
<!-- This is invalid -->
<div onClick={...}>Update</div>

<!-- This is better -->
<button onClick={...}>Update</button>
```



And that's the end of content!

Use accessibility tools

Apple

Apple's accessibility tools come out of the box, both for macOS and iOS. Please have a go at using VoiceOver:

- Guide for VoiceOver on Mac
- Guide for VoiceOver on iPhone

Chrome

For all computers including Windows and Linux, you can use the Chrome Screen Reader.

Android

TalkBack is the Android screen reader.

Additional topics

- **Reduced motion**
  - MDN article on prefers-reduced-motion
- **Dark mode**
  - Dark mode can also improve accessibility
- **Audio descriptions**
  - Example of engaging audio descriptions

All the best!