

# Accessibility in the Browser

# Cassidy Williams

---

Software Engineer & Developer  
Evangelist at Venmo

[cassidoo.co](http://cassidoo.co)

# The internet is great.

---

<http://www.noooooooooooooo.com/>

If you can see the screen.

If you can use the mouse to  
click a button.

If you can hear the audio  
that plays when you click  
it.

That is, if you have no  
disabilities of any kind.

The web has amazing potential for people with disabilities.

**Most studies find that about 20% of  
the population has some kind of  
disability.**

So, why aren't we  
accounting for them?

# What we're gonna talk about today

- The Semantic Web
- WAI-ARIA Specification
- Accessibility Checklist for your projects
- Preguntas

# The Semantic Web

Semantics is the science of  
meaning

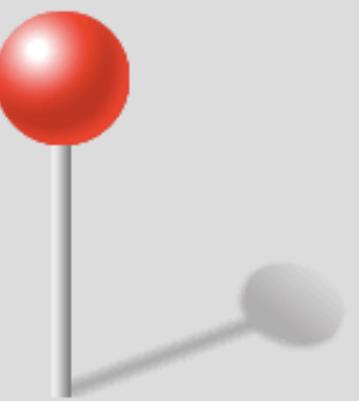
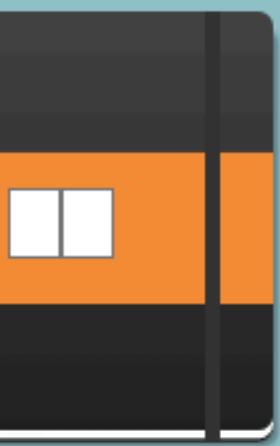
# The Semantic Web

- HTML5 Tags
- The role attribute

# HTML5 Semantic Tags



Probably the easiest thing you can do to make your site more accessible is to use them.



These possibilities are  
great.

But let's look at  
a.singlediv.com with a  
screen reader.

# The role attribute

Provides a programmatically determinable relationship between specific content.

  
<p>Lola has been up to no good.</p>



Lola has been up to no good.

```
<figure role="group">
  
  <figcaption>Lola has been up to no good.</figcaption>
</figure>
```



Lola has been up to no good.

Adding a role will not make an element act or look differently for people not using assistive technology.

# The WAI-ARIA Specification

---

W3C's Accessible Rich Internet  
Applications Spec

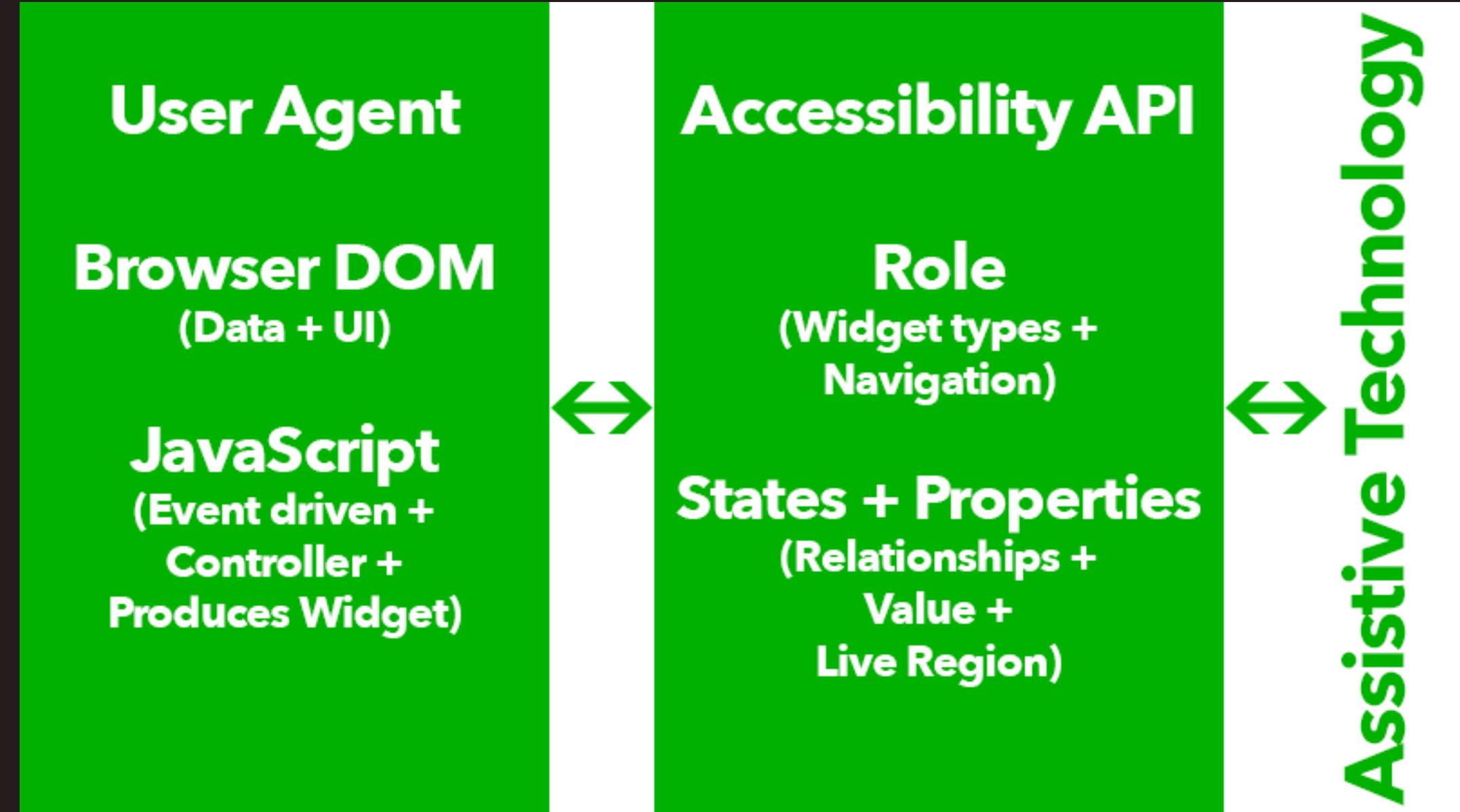
# The WAI-ARIA Specification

---

Defines ways to make Web content and Web applications more accessible to people with disabilities.

# The Contract Model with Accessibility APIs





There are 4 rules for ARIA  
use.

# Rule #1

If you can use a native HTML element or attribute, then do so.

# Rule #2

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

If you want your heading to be a button, do this:

```
<h1><button>This is a heading AND a button</button></h1>
```

And not this:

```
<h1 role=button>This is a heading AND a button</h1>
```

# Rule #3

All interactive ARIA controls must be usable with the keyboard.

# Rule #4

All interactive elements must have an accessible name.

# Bad: An input with a label

```
<label>Favorite Taco Brand</label> <input type="text">
```

# Good: An input with a label with a name

```
<label for="taco">Favorite Taco Brand</label> <input type="text" id="taco">
```

# ARIA Examples

**aria-describedby**

Specifies a hyperlink to content that describes the object.

# aria-describedby

```

```

```
<div id="lola-desc">Lola has been up to no good.</div>
```



Lola has been up to no good.

# aria-describedby

```

```

```
<div id="lola-desc">Lola has been up to no good.</div>
```



# University of Illinois' Example

<http://test.cita.illinois.edu/aria/toolbar/toolbar1.php>

# Roles:

- role="application"
- role="button"

# States and properties:

- aria-controls
- aria-pressed
- aria-labelledby

# Accessibility Checklist for your projects

# Visibility

- Use the visibility or display properties, or the hidden attribute
- Don't override too many of the browsers' default behaviors

# Color

---

There should be a certain amount of contrast in your color choices.

# Focusability

- Everything on the page should be able to have focus
- If a control on the page isn't standard, they should have an appropriate ARIA role attached to them

# State Changes

---

Standard controls have this taken care of, but for custom controls, ARIA states provide solutions (`aria-checked`, `aria-pressed`, etc)

# Developer Tools

Fang's Screen Reader Emulator

WAI-ARIA Best Practices

Color Contrast Checker

Tanaguru's Automated Accessibility Testing Service

<http://www.html5accessibility.com/>

Let's make the web  
accessible by everyone.

# Cassidy Williams



[cassidoo.co](http://cassidoo.co)

[github.com/cassidoo](https://github.com/cassidoo)