

Standardizing `<select>`:

What the future holds for HTML Controls

Hi! I'm Stephanie.

Program Manager for
Microsoft Edge
Developer Experiences

Twitter: @seaotta

First name:

I

Last name:

I

email:

I

☐ Male

☐ Female

Send

Reset

First name*

Last name*

DOB*

Required by Public Health Authorities

Sex*

Required by Public Health Authorities

Home Address*

[Enter address manually](#)

Email*

We will send your test details by both SMS/Email

Confirm email*

Mobile number*


We will send your test details by both SMS/Email

Confirm mobile number*

<input type="date">

`<input>` elements of `type="date"` create input fields that let the user enter a date, either with a textbox that validates the input or a special date picker interface.

The resulting value includes the year, month, and day, but *not* the time. The [time](#) and [datetime-local](#) input types support time and date+time input.

 HTML Demo: <input type="date">Reset

HTML

CSS

```
1 <label for="start">Start date:</label>
2
3 <input type="date" id="start" name="trip-start"
4     value="2018-07-22"
5     min="2018-01-01" max="2018-12-31">
6
```

Output

Start date:

May 2018 ▾

2014

2015

2016

2017


2018

| | | | |
|-----|-----|-----|-----|
| Jan | Feb | Mar | Apr |
| May | Jun | Jul | Aug |
| Sep | Oct | Nov | Dec |

The input UI generally varies from browser to browser; see [Browser compatibility](#) for further unsupported browsers, the control degrades gracefully to `<input type="text">`.


Value

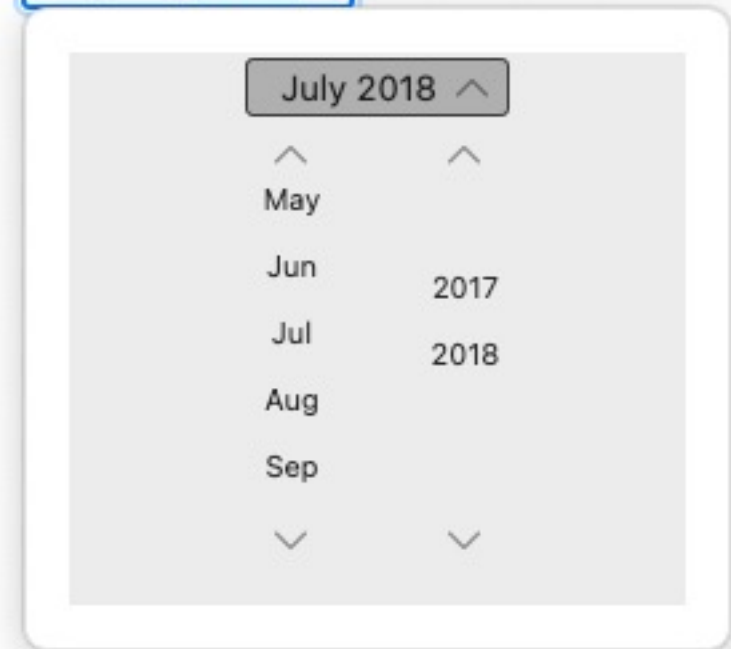
A [DOMString](#) representing a date in YYYY-MM-DD format, or

Start date: 07 / 22 / 2018 



Chromium

Start date: 07 / 22 / 2018 



Firefox

Start date: 07/22/2018

July 2018 ▾

| |
|------|
| 2014 |
| 2015 |
| 2016 |
| 2017 |
| 2018 |

| | | | |
|-----|-----|------------|-----|
| Jan | Feb | Mar | Apr |
| May | Jun | Jul | Aug |
| Sep | Oct | Nov | Dec |

Chromium

Date picker label

mm/dd/yyyy

< **October 2017** ▴ ▾ >

| S | M | T | W | Th | F | S |
|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| 29 | 30 | 31 | 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |

IBM's Carbon Design System

 A brief overview of the past of HTML Controls

 Current problems with native HTML Controls

 How we're proposing solving these pain points

Some History

HTML Form Controls

1995

Hypertext Markup Language - 2.0

September 22, 1995

T. Berners-Lee (1)

D. Connolly (1)

(1) MIT/W3C

- [Forms](#)
 - [Form Elements](#)
 - [Form: FORM](#)
 - [Input Field: INPUT](#)
 - [Text Field: INPUT TYPE=TEXT](#)
 - [Password Field: INPUT TYPE=PASSWORD](#)
 - [Check Box: INPUT TYPE=CHECKBOX](#)
 - [Radio Button: INPUT TYPE=RADIO](#)
 - [Image Pixel: INPUT TYPE=IMAGE](#)
 - [Hidden Field: INPUT TYPE=HIDDEN](#)
 - [Submit Button: INPUT TYPE=SUBMIT](#)
 - [Reset Button: INPUT TYPE=RESET](#)
 - [Selection: SELECT](#)
 - [Option: OPTION](#)
 - [Text Area: TEXTAREA](#)
 - [Form Submission](#)
 - [The form-urlencoded Media Type](#)
 - [Query Forms: METHOD=GET](#)
 - [Forms with Side-Effects: METHOD=POST](#)
 - [Example Form Submission: Questionnaire Form](#)

1995

First name:

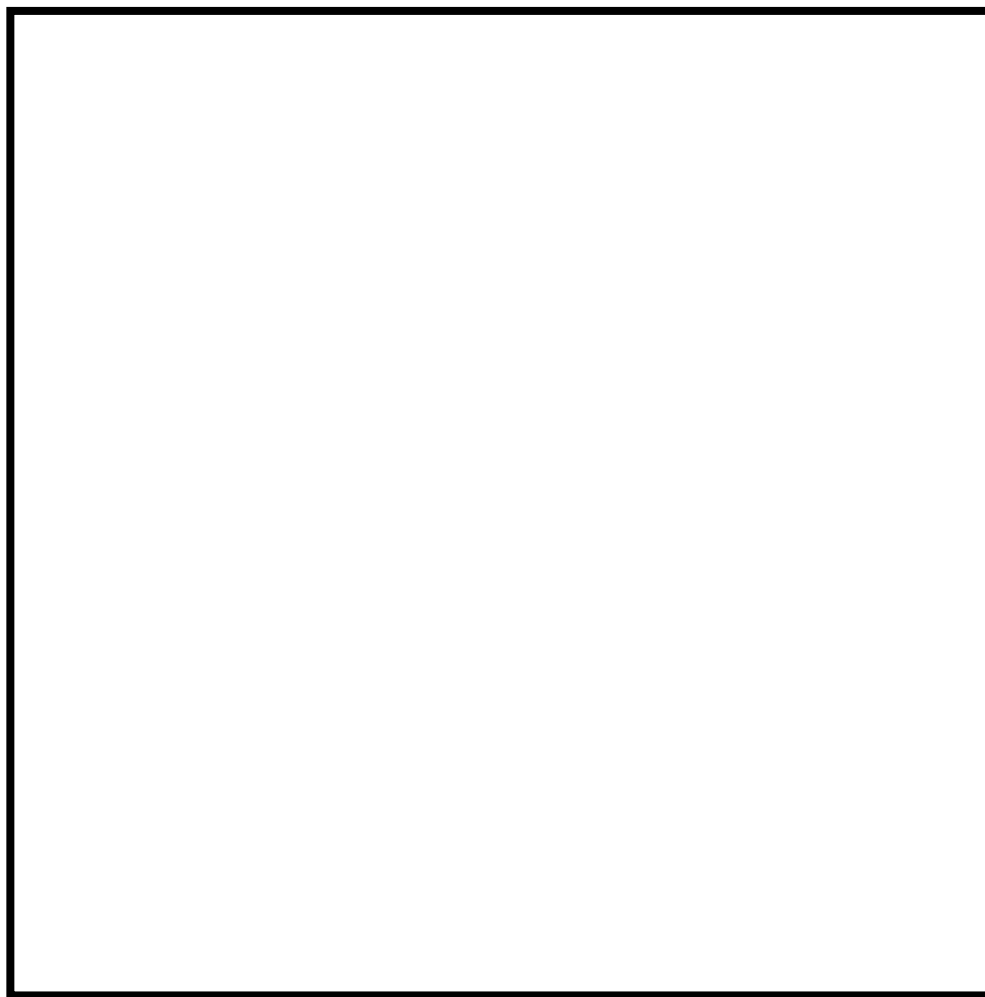
John

Last name:

Doe

Submit

1995



CSS

IS

AWESOME

1997

CSS

IS

AWESOME

1999

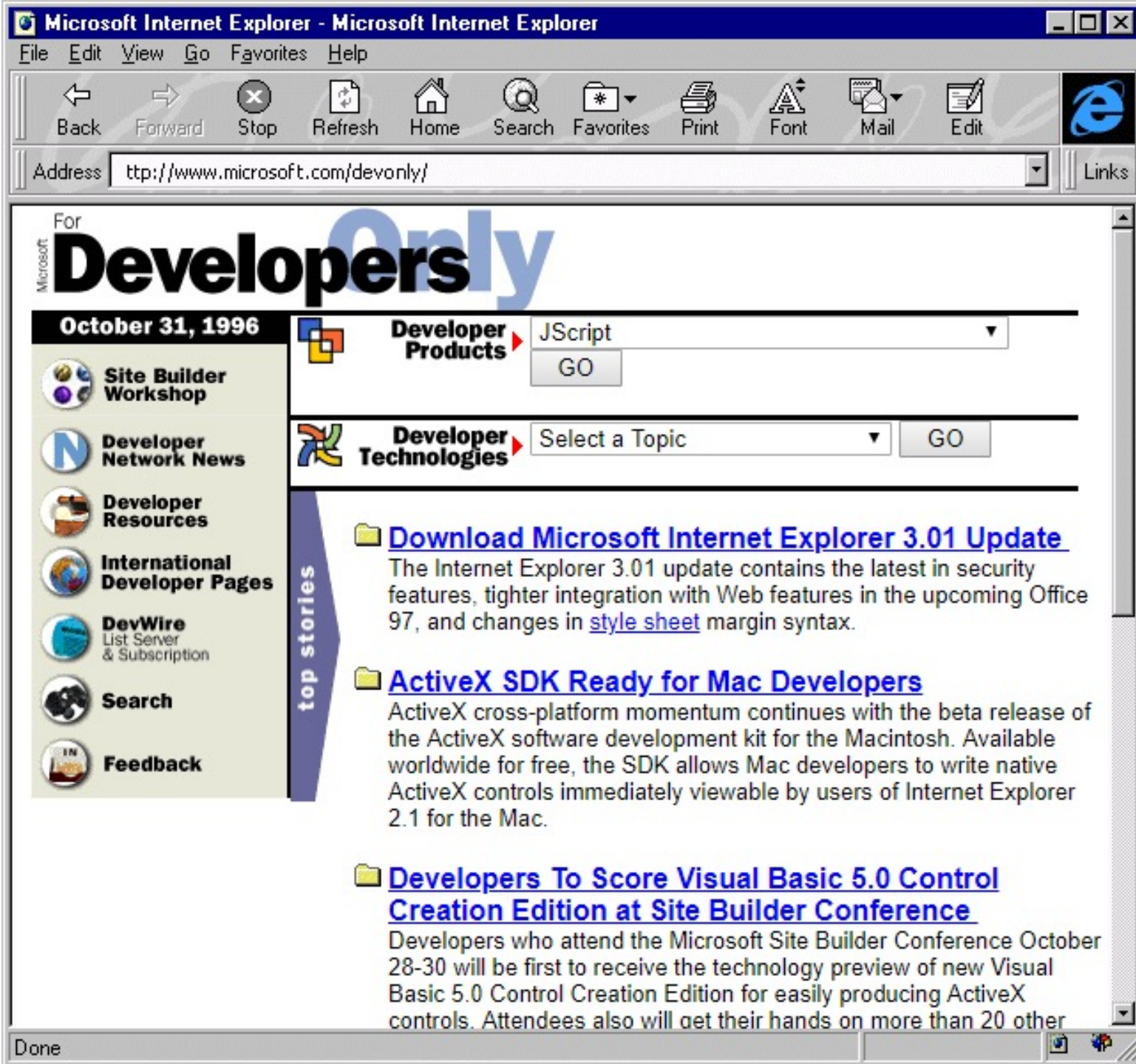
<HTML>

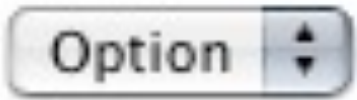
4.01



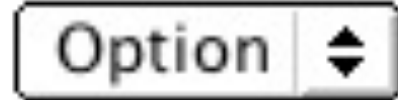
CSS
IS
AWESOME

1995

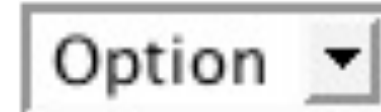




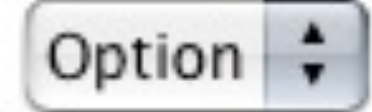
Safari



IE Mac



Moz/FF Mac



Opera Mac



Camino



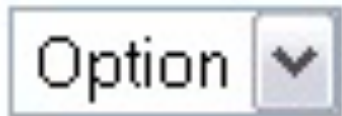
IE Win 2k



Moz/FF Win 2k



Opera Win 2k



IE Win XP



Moz/FF Win XP



Opera Win XP

```
.form {  
    -webkit-appearance: value;  
    -moz-appearance: value;  
    appearance: value;  
}
```

Can I use

? ⚙ Settings

x | Feature: CSS Appearance

CSS Appearance 📄 - WD

The **appearance** property defines how elements (particularly form controls) appear by default. By setting the value to **none** the default appearance can be entirely redefined using other CSS properties.

| Usage | % of | all users | ? |
|-------------|------|-----------|----------|
| Global | 0% | + 95.39% | = 95.39% |
| unprefixed: | 0% | + 2.04% | = 2.04% |

| Current aligned | | Usage relative | | Date relative | | Apply filters | | Show all | | ? | | | | |
|-----------------|--------------------|-------------------|-------------------|----------------------|-------------------|----------------------|--------------|-----------------------|----------------|--------------------|---------------------|------------------------|-------------------|-----------------|
| IE | Edge * | Firefox | Chrome | Safari | Opera | iOS Safari * | Opera Mini * | Android Browser * | Opera Mobile * | Chrome for Android | Firefox for Android | UC Browser for Android | Samsung Internet | QQ Browser |
| | | <div>132-34</div> | | | 10-12.1 | | | | | | | | | |
| 6-10 | <div>1212-18</div> | <div>135-71</div> | <div>14-78</div> | <div>13.1-12.1</div> | <div>115-63</div> | <div>13.2-13.1</div> | | <div>12.1-4.4.4</div> | 12-12.1 | | | | <div>14-9.2</div> | |
| 11 | <div>179</div> | <div>172</div> | <div>179</div> | <div>113</div> | <div>164</div> | <div>113.2</div> | all | <div>176</div> | <div>146</div> | <div>179</div> | <div>168</div> | <div>112.12</div> | <div>110.1</div> | <div>11.2</div> |
| | | <div>173-74</div> | <div>180-82</div> | <div>1TP</div> | | <div>113.3</div> | | | | | | | | |

Notes Known issues (0) Resources (2) Feedback

- ¹ The appearance property is supported with the **none** value, but not **auto**. WebKit, Blink, and Gecko browsers also support additional vendor specific values.
- ² Microsoft Edge and IE Mobile support this property with the `-webkit-` prefix, rather than `-ms-` for interop reasons.
- ³ `-moz-appearance:none` doesn't remove the dropdown arrow in select tag

1995 – present

The Current State of Styling Native Controls

Feasibility of Styling Form Controls with CSS

Can be styled with few problems

<form>*

<fieldset>

<label>

<output>

Text-field (<input>)*

Buttons

Can be styled with complex CSS and hacks

Checkboxes*

Radios*

<legend>

Good night and good luck.

<select>*

<option>*

<optgroup>

<datalist>

<progress>

<meter>

Color picker

Date controls

Dropdown widgets

Range

File picker

*** Original Forms introduced in 1995**

Browser Inconsistencies

Browser Inconsistencies

Chrome

No file chosen

EdgeHTML

Firefox

No file selected.

Safari

no file selected

No extensibility



Scott Jehl

@scottjehl



you have one problem:
you want icons in your `<select>` menu options.

you decide to make a custom select menu:
you now have at least 75 problems.

11:50 AM · Feb 6, 2020 · [Twitter Web App](#)

41 Retweets **199** Likes

```
<video controls width="1080">  
</video>
```



0:00



```
<video controls width="1080">  
</video>
```

▶ 0:00



```
<video width="1080">  
</video>
```

UI Components

Dropdown

↓

Secondary

Radio Group

Jotaro Kujo

Joseph Joestar

Dio Brando



Value Slider

At least 8,000

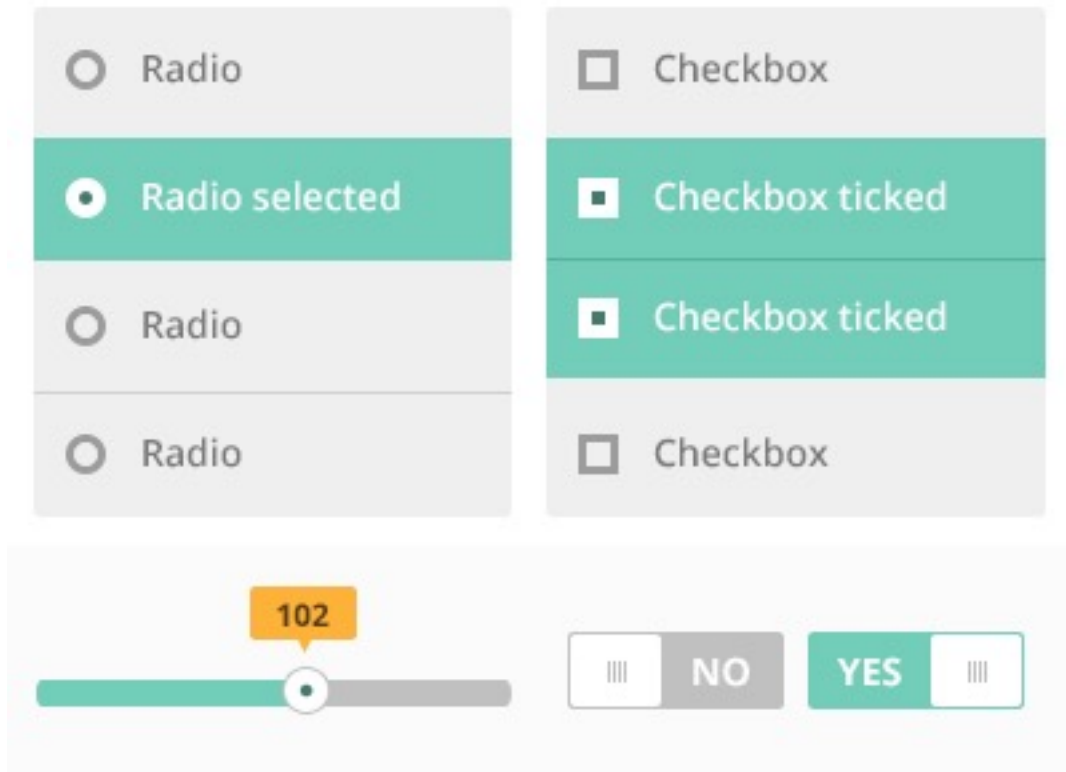


Value Range

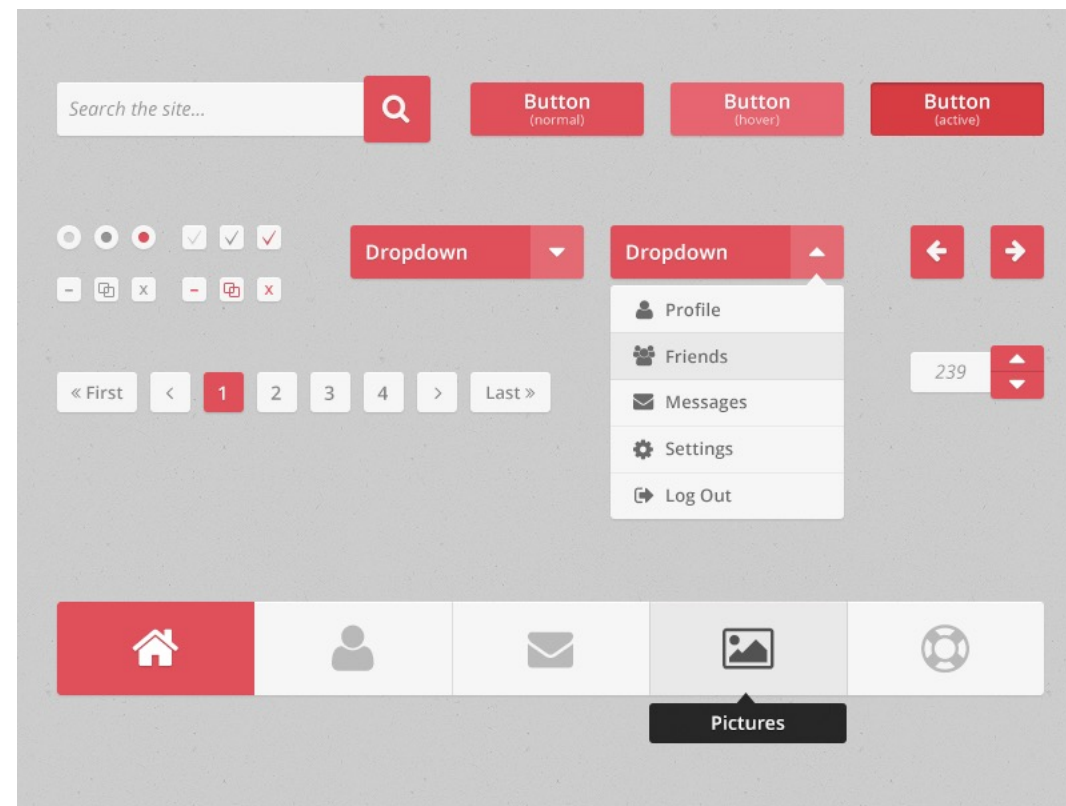
Between 4,000 and 9,000



Dribbble: Jordan Ranson



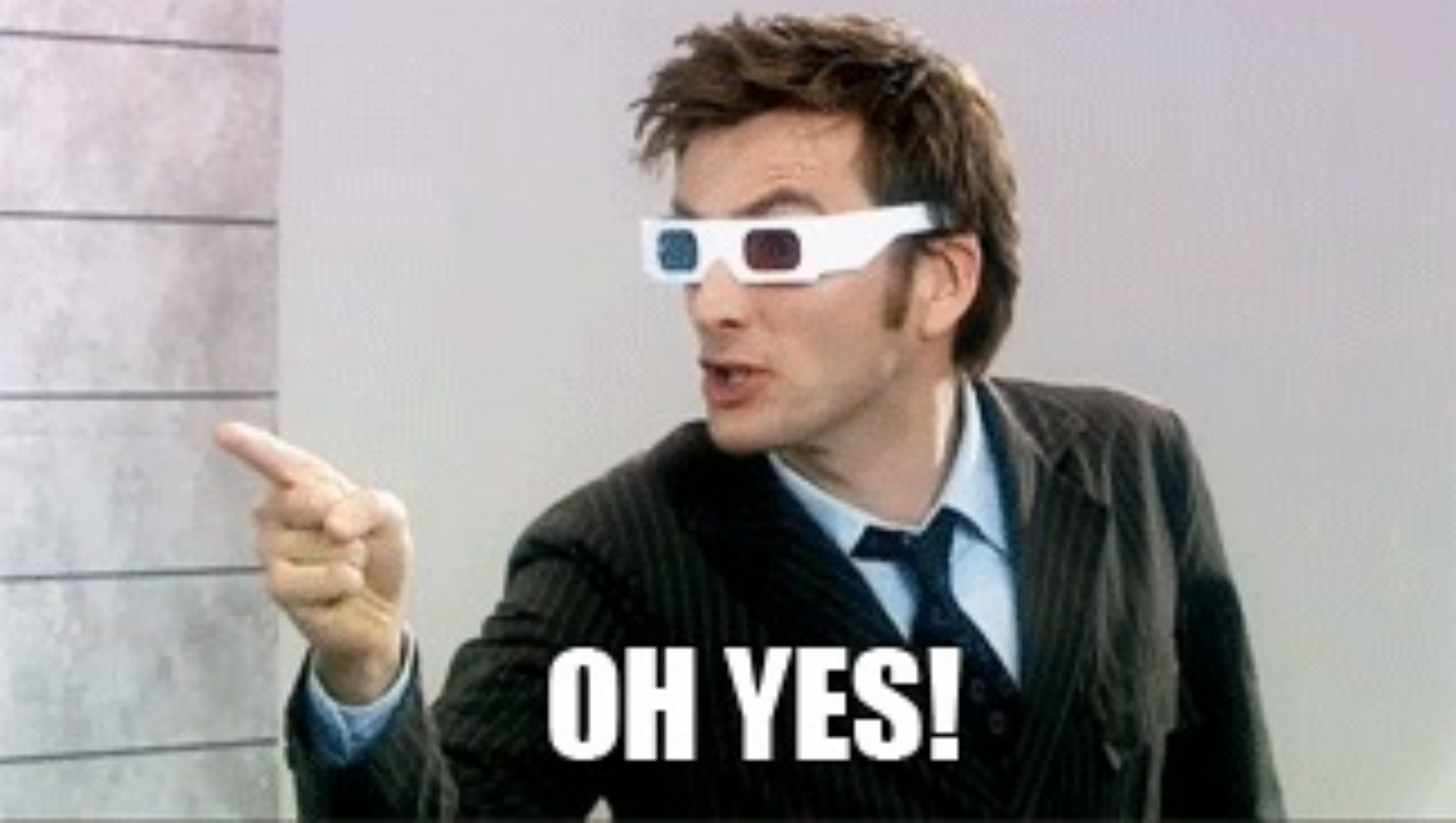
Dribbble: Mark Hendriks



Dribbble: Seb kay

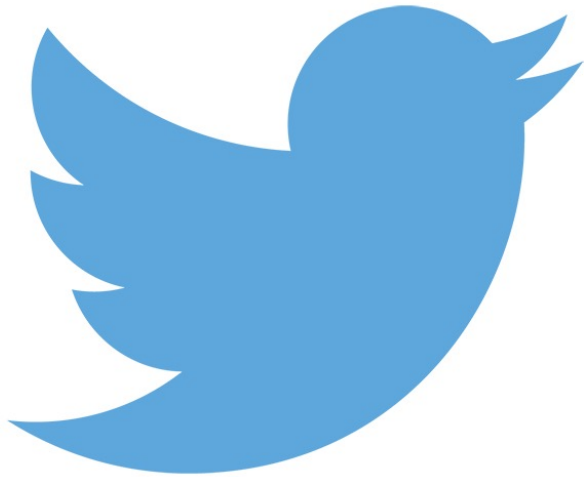
What do
developers
really want?

Do they really
want better
form controls?



OH YES!

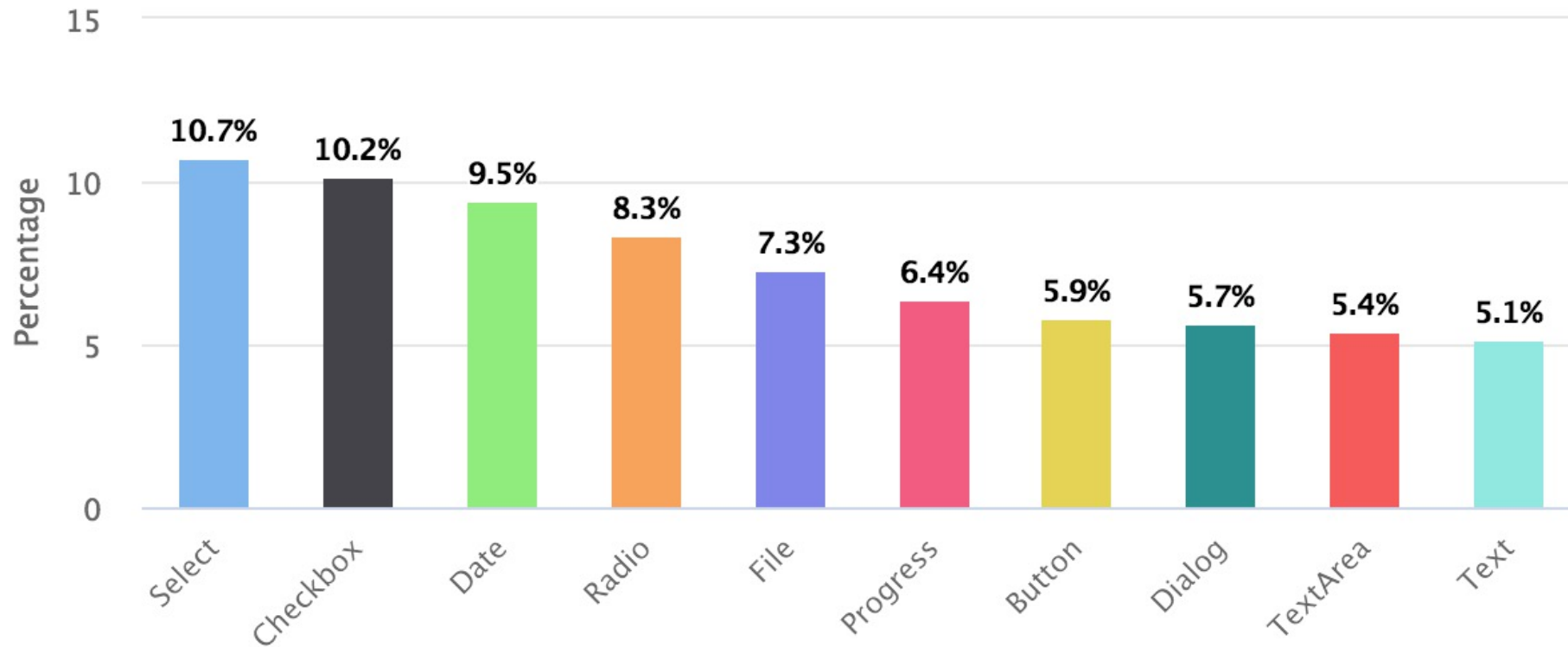
Initial Research



1400 respondents

**Full stack, front-end, back-end,
designers, and other**

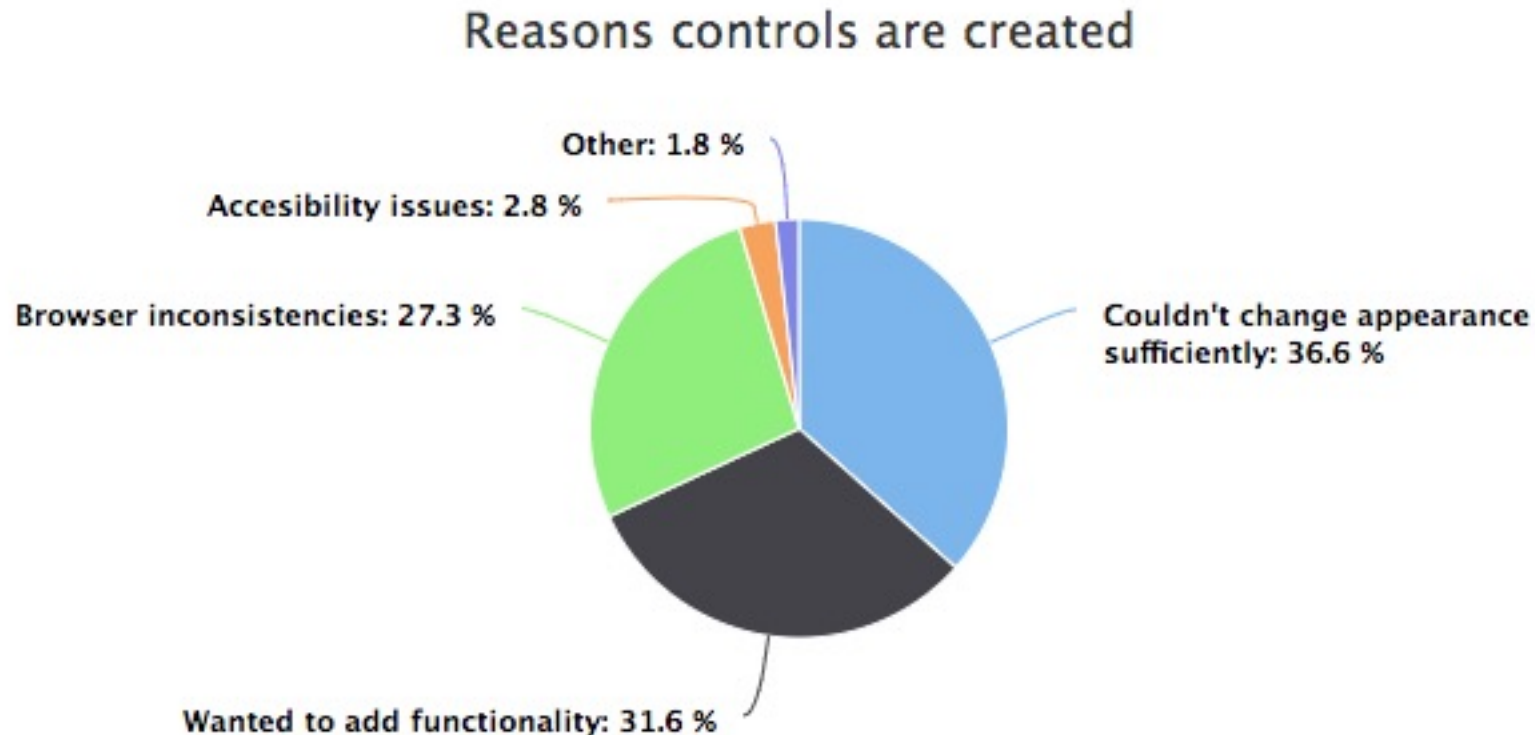
Top 10 Re-created Form Controls



Source: gwhitworth.com

Full results: <https://aka.ms/controls-survey>

Reasons Controls Are Created



Source: gwhitworth.com

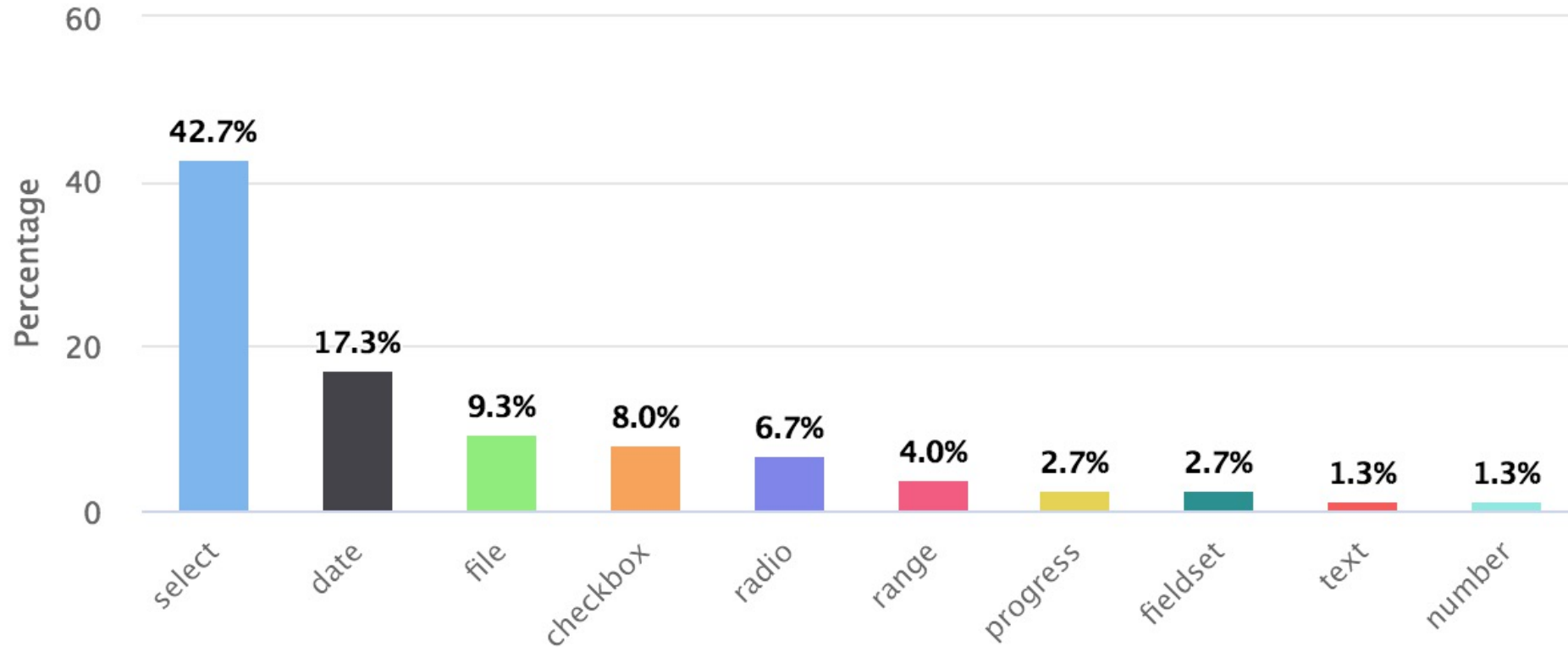
JSConfEU Survey

Which form control gives you the most frustration?

Why?

JSConfEU Survey

Which form control gives you the most frustration? Why?



Source: gwhitworth.com



Requires
hacky tricks



Can't
style `<option>`
elements at all
to the extent we
need to



**...but the amount of
work it takes to
implement an
accessible alternative
with complete feature
parity is massive.**

How painful is it?

Important Research



Stephanie Stimac 🧙 Casting Spells
@seaotta



Dear devs and front-end designers, important research for a talk I'm giving: please fill in the blank:

"I would rather _____ than attempt to style a native `<select>` element"

6:40 PM · Jan 6, 2020 · [Twitter Web App](#)





dilski  @dilski · Jan 7

Replying to @seaotta

Call each person attempting to use the form and ask them what option they would like

1



36



Hagbard Celine @asciidisco · Jan 6

Replying to @seaotta

Convince my fellow programmers to reinstall Silverlight.



2



Martin Adams @Martin_Adams · Jan 6

Replying to @seaotta

Build the entire site in Flash

2



212



Ed Henderson @edhenderson · Jan 6

Replying to @seaotta

Support IE6



1



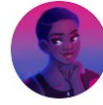
marty  @Marty331 · Jan 7

Replying to @seaotta

I would rather use vim.



1



Keziah @KeziahL · Jan 7

Replying to @seaotta

Breakdance barefoot on a pile of Legos

1



12



Ben Horst @benhorst · Jan 6

Replying to @seaotta

Grind my teeth to the jawbone with rough sandpaper



1



Iarf @Iarf2k · Jan 7

Replying to @seaotta

chew on glass



1



Chelsea Adams @thecityinspeech · Jan 7

Replying to @seaotta

Boil my toes in lava



1



Mike Francis @_mikefrancis · Jan 6

Replying to @seaotta

Maybe a bit melodramatic but heat up a rusty old fork, with a few tines (yes I had to google that) snapped off and broken, then with both arms thrust it into my inner thigh



2



**Clearly, there is
an issue**

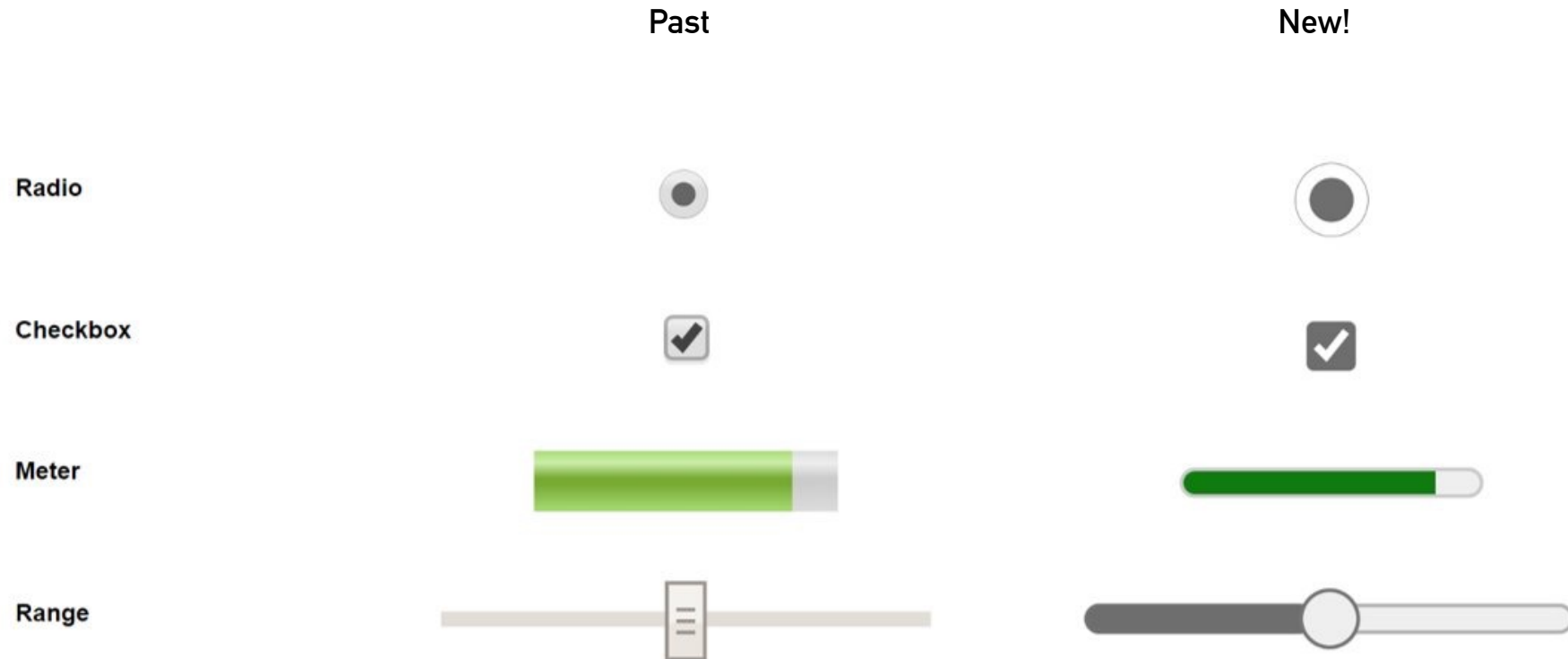
What's happened and
what's to come





New Styles & Accessibility Improvements

Chromium Controls Visual Refresh



Dark Mode for HTML Form Controls

Dark Mode for HTML Form Controls

☐ ☒

☐ ☒

Button

Text value here Placeholder here

xxxx

4

Text value here

value 1 ▾

value 1 ▴
value 2
value 3 ▾

mm/dd/yyyy Week --, ----

--:-- -- mm/dd/yyyy --:-- --

Choose File No file chosen

value

[color-scheme_meta value](#)

☐ ☒

☐ ☒

Button

Text value here Placeholder here

xxxx

4

Text value here

value 1 ▾

value 1 ▴
value 2
value 3 ▾

mm/dd/yyyy Week --, ----

--:-- -- mm/dd/yyyy --:-- --

Choose File No file chosen

value

[color-scheme_meta value](#)

```
@media (prefers-color-scheme: dark) {  
    // styles here  
}
```

```
<meta name="color-scheme" content="light dark" />
```

Available in:

- Microsoft Edge on the desktop (v87)
- Chrome for Android (v91)
- Future versions of Microsoft Edge for Android

[**https://aka.ms/dark-controls**](https://aka.ms/dark-controls)

HTML isn't done

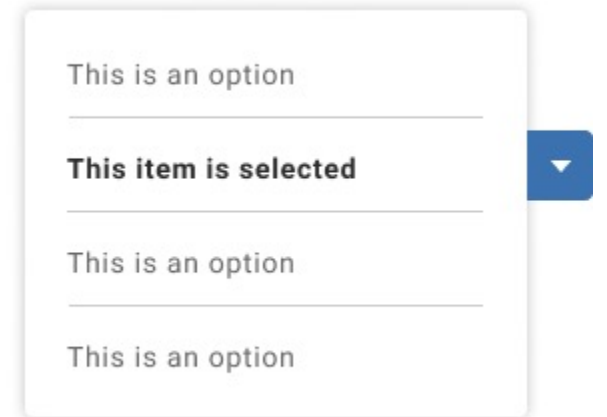
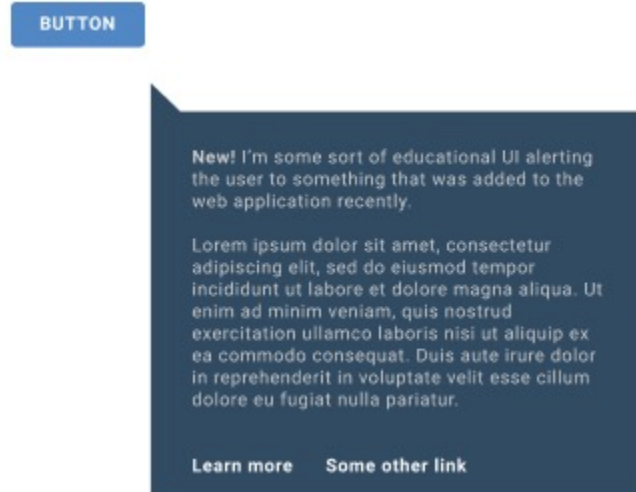
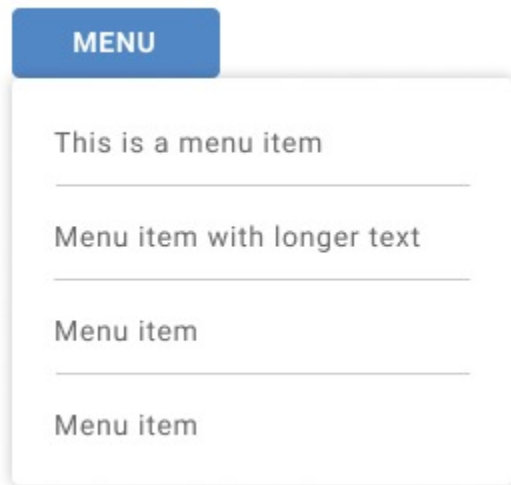
New Native Elements

New proposal: <popup> element

Microsoft / Google / Salesforce

Transient, top-layer UI

Popover elements which light dismiss, typically only one is open at a time (with exceptions for child/chained popups)



**We propose that all popups
share “light dismiss” behaviors.**

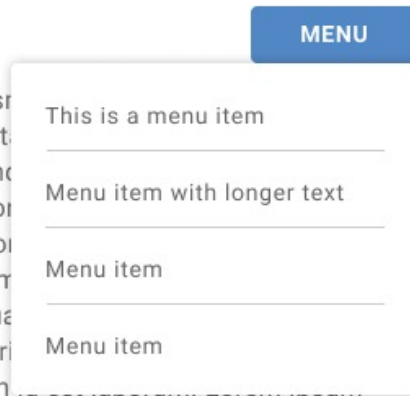
Elements that don't have light dismiss behavior

- Alerts
- Toasts
- Custom tooltips
- Miscellaneous, persistent popover UI

Basic Example

Section Title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



```
<button aria-haspopup="true" aria-controls="menuPopup" id="menuButton">Menu</button>
<popup id="menuPopup" role="menu" anchor="menuButton">
  <!-- Markup for menuitems goes here -->
</popup>

<script>
  document.getElementById('menuButton').addEventListener('click', () => {
    document.getElementById('menuPopup').show()
  });
</script>
```

[**https://aka.ms/popup-explainer**](https://aka.ms/popup-explainer)

Other controls +
controls-related
work in progress

Showing off Spicy Sections

This demo shows different designed affordances at different screen sizes (view the source too)...

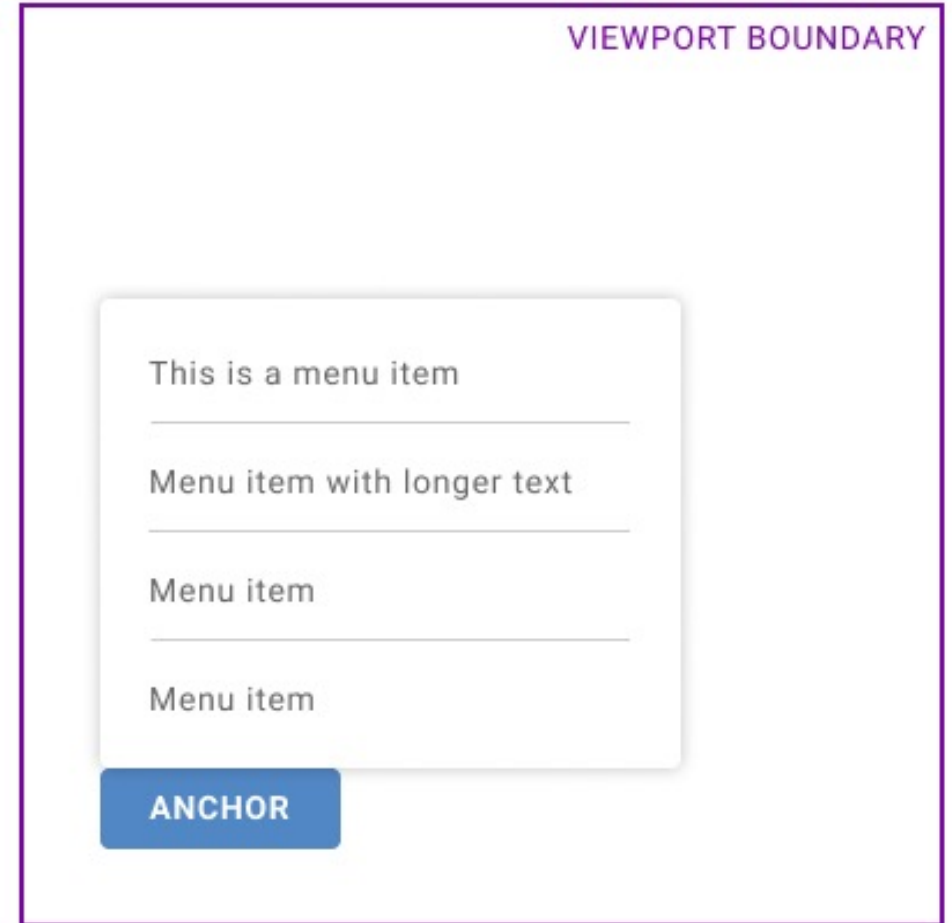
- `0-30em`: collapsable sections
- `30-60em`: structured content
- `60em+`: tab-bar

Tabs **Collapse** **Structured Content**

The fallback state is simply structured HTML, without any show/hide affordances. This does not require any special styling.

[**https://aka.ms/tabs-demo**](https://aka.ms/tabs-demo)

Anchored Positioning



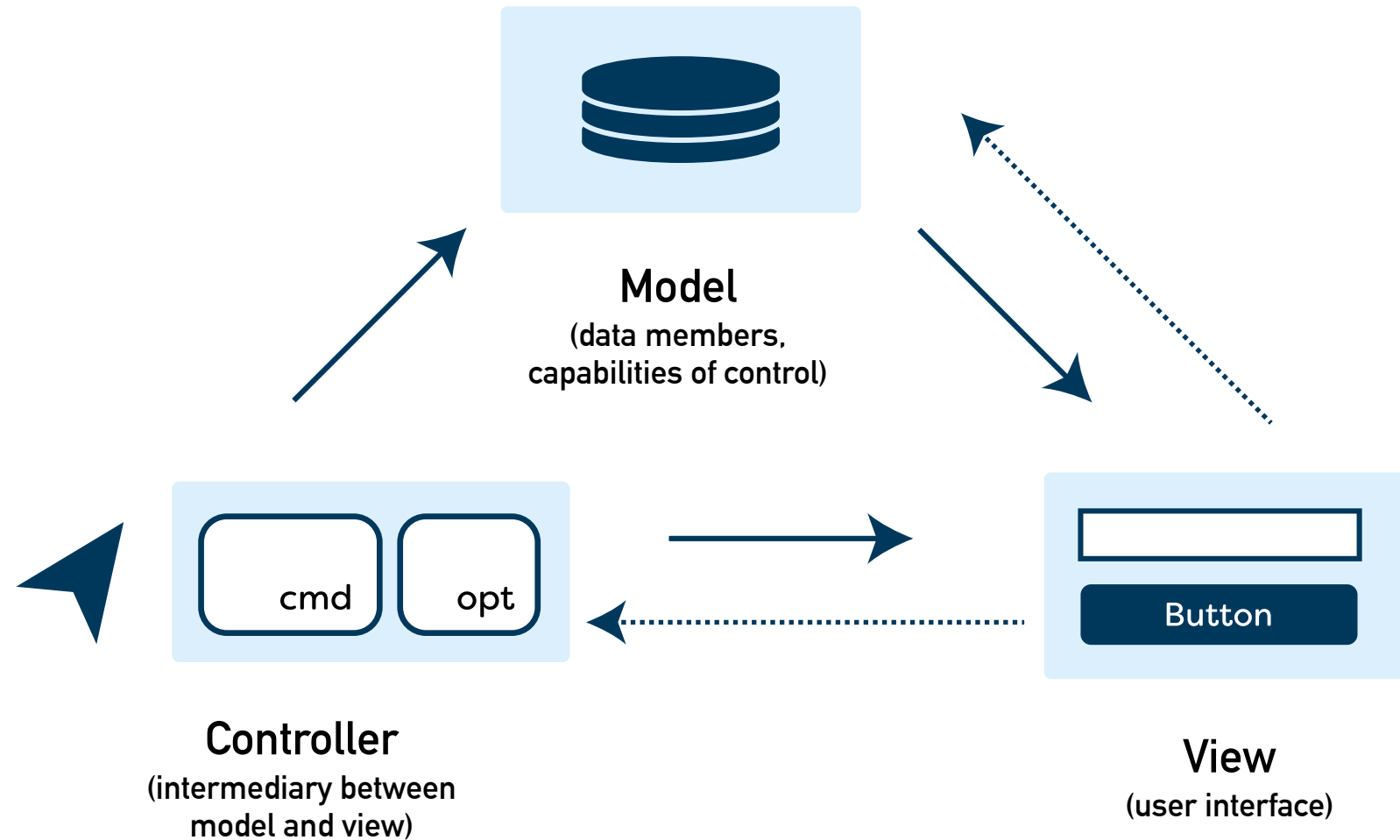
[**https://aka.ms/anchor-pos**](https://aka.ms/anchor-pos)

Fixing the current
problems with controls

Customizing Controls UI

Customizing Controls UI Explainer

MVC Design Pattern



Goals

- Style any arbitrary part of a native control
- Add arbitrary content into a control
- Style a particular part without rewriting the whole UI
- Customize UI without reimplementing the data model and code for reacting to user input
- Customized controls are accessible by default

3 Customization Options

- Using standardized parts and states to override the native control styles via pseudo-classes and pseudo-elements.
- Using named <slots> to replace parts of the native control UI with developer content, while optionally leaving some native parts in place.
- Replacing the entire UI of the native control with an author-supplied shadow root.

Solution 1:

**Standardized
Control UI anatomy, parts
and behavior**

Solution 1:

**Standardized
Control UI
anatomy,
parts and
behavior**



aka.ms/open-ui

Goals of Open UI

- Document component names as they exist today
- Define a common language for describing UIs and design systems
- Eventual browser standards for web app components
- Converging designer processes and developer workflows

<select>
investigation

[Home](#)[Charter](#)[Contribute](#)[Analysis](#)[Button](#)[Card](#)[Component Name Matrix](#)[Datepicker](#)[Skeleton](#)[Slider](#)[Switch](#)[Text](#)[Components](#)[Checkbox \(Working Draft\)](#)[Table \(Editor's Draft\)](#)[Select \(Editor's Draft\)](#)

Select (Editor's Draft)

[View Research](#)

Overview

The `<select>` is a control that provides a list of options for the user to select from.

Use Cases

The `<select>` control is primarily leveraged to select an option for within a form. For example when your buying a shirt you may be provided with a `<select>` that has options for sizes that you then select the appropriate one for you.

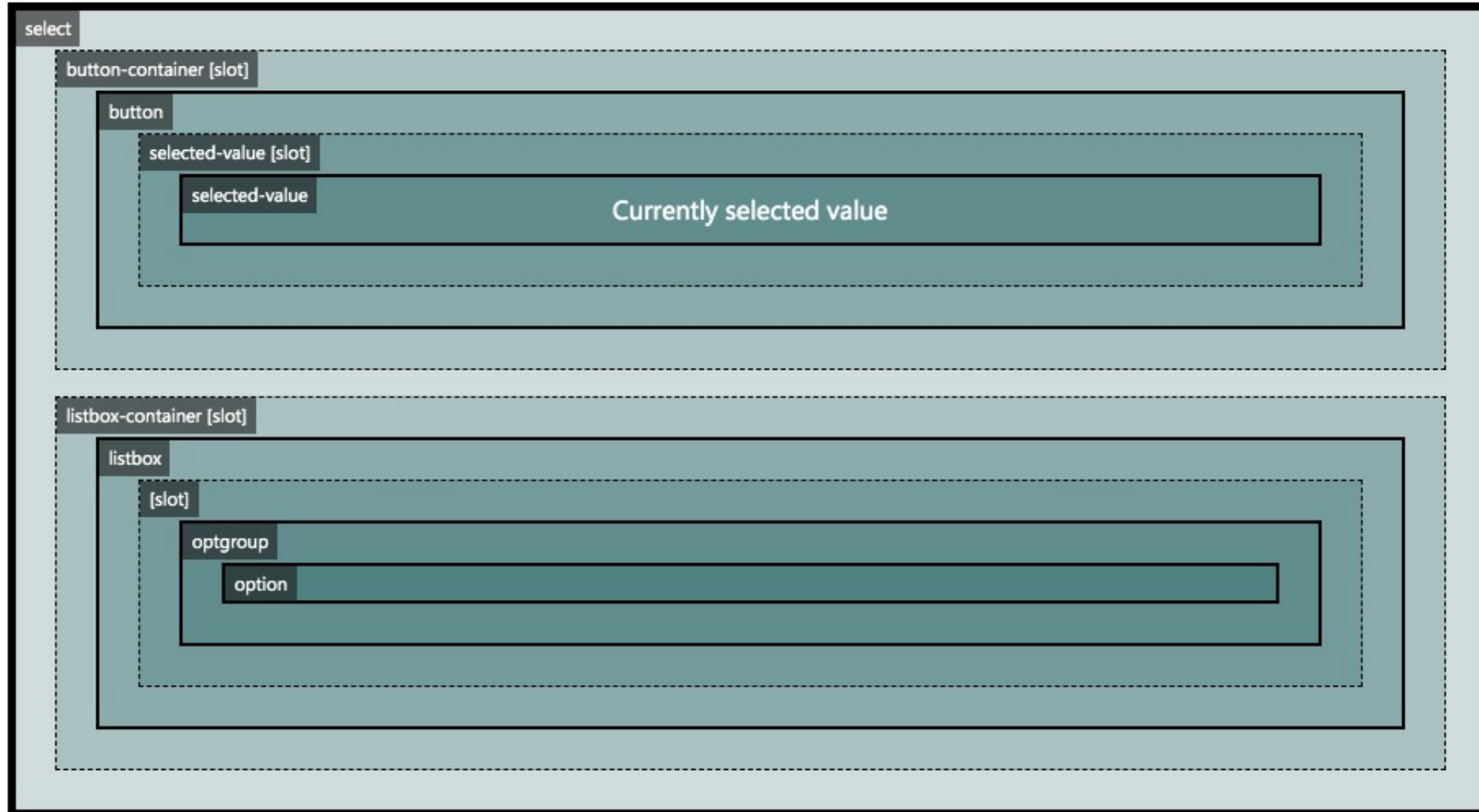
Prior Art/Examples

- [Ant Design](#)
- [Carbon Design System](#)
- [FAST DNA](#)
- [Atlas Kit](#)

`<select>` Properties

| Attribute Name | Type | Default Value | Description |
|---------------------------|---------------------|-----------------------------------|--|
| <code>autocomplete</code> | <code>string</code> | <code>off</code> | Allows the developer to provide a hint on how to search the content within the <code><option></code> (s) |
| <code>autofocus</code> | <code>bool</code> | <code>false</code> | If set to <code>true</code> the <code>input</code> will have focus set on page load |
| <code>disabled</code> | <code>bool</code> | inherited from containing element | If set to <code>true</code> the user will not be able to interact with the control |
| <code>form</code> | <code>string</code> | the form that contains the | Represents the form owner, via the <code>form</code> attribute being |

The anatomy of a `<select>`



```
<style>
  .styled-select::part(button) {
    background-color: red;
  }
</style>
```

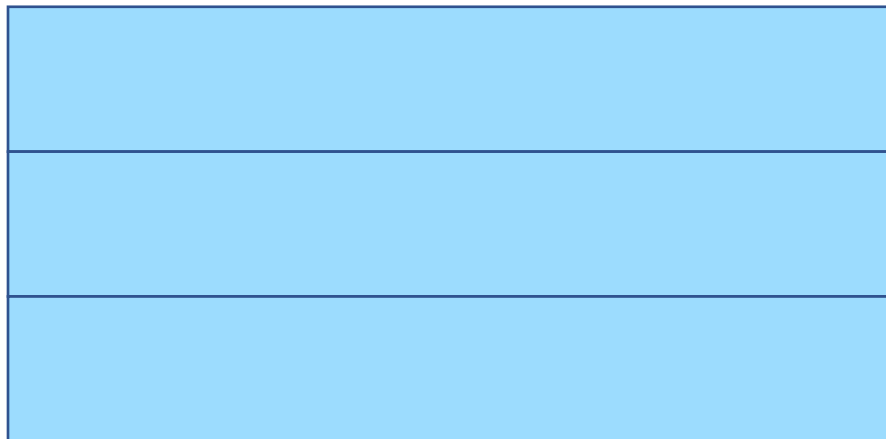
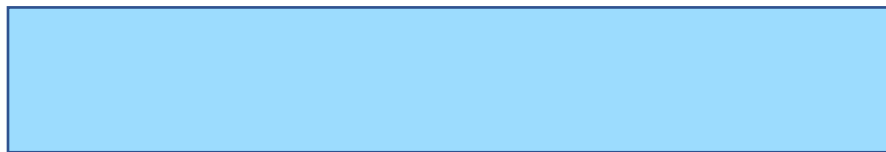
```
<select class="styled-select">
  <option>choice 1</option>
  <option>choice 2</option>
</select>
```

```
<style>
  .styled-select:open::part(button) {
    background-color: lightgray;
  }
</style>
```

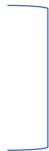
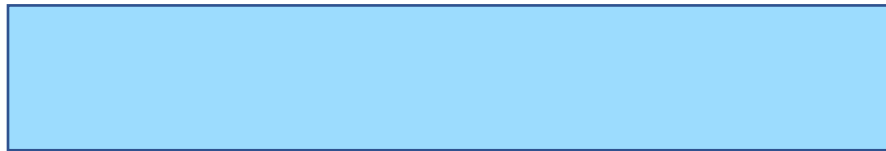
Solution 2:

Named Slots

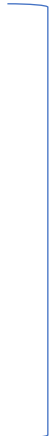
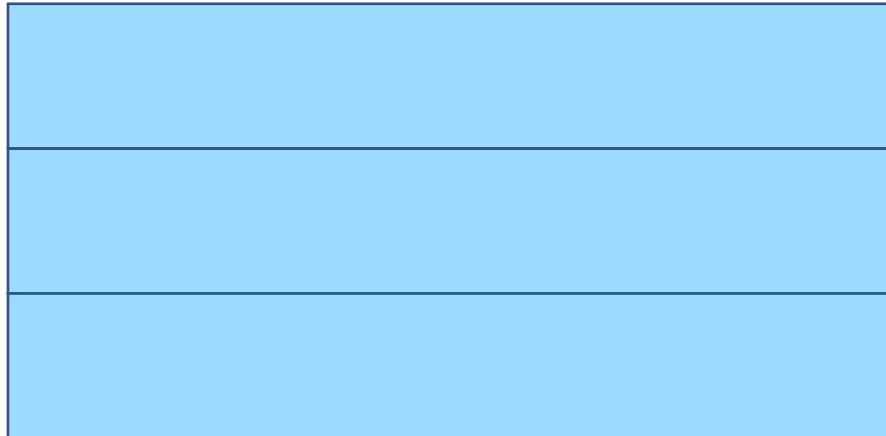
Standardized Slots



Standardized Slots



slot=button



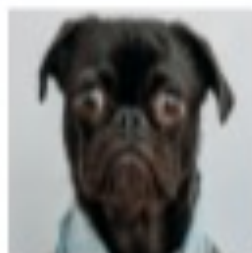
slot=listbox

```
<select>
  <div slot="button" part="button" class="custom-button">Choose a pet</div>
  <div slot="listbox" part="listbox" class="custom-listbox">
    <option>
      
      <div class="option-text">Cat</div>
    </option>
    <option>
      
      <div class="option-text">Dog</div>
    </option>
  </div>
</select>
```

Choose a pet



Cat



Dog

Customize selectively

```
<input type="range">
```

Customize selectively

```
<range>  
  <div slot="thumb" part="thumb">  
    <svg><!-- Use SVG to draw the thumb icon... --></svg>  
  </div>  
</range>
```

Parts and slots

Native controller code + user-provided UI Parts

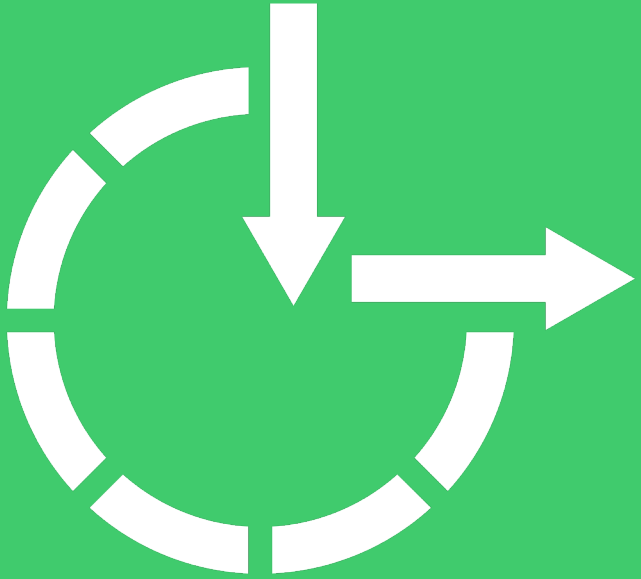
Web Platform
Code

+

Native event
handlers

+

Part
Attributes /
element types



Accessibility

Solution 3:

Shadow DOM replacement

attachShadow()

~~attachShadow()~~

```
let customSelect = document.createElement('select');
customSelect.setAttribute("custom", "");
let selectShadow = customSelect.attachShadow({ mode: 'open' });
selectShadow.innerHTML = `My custom select UI`;
document.body.appendChild(customSelect);
```

Don't forget your parts!

```
part="button"
```

<https://aka.ms/controls-explainer>

We need you!

We need you!

- Contribute to the form control investigations on Open UI
- Tell browser vendors what you need from your form controls
- Provide feedback on the explainers

Follow these folks

- @gregwhitworth – Salesforce / Open UI
- @stubbornella – Google Chrome PM
- @seaotta – Microsoft Edge PM

We're here to listen...

...because these
improvements are for you.

Thank you!

<https://noti.st/seaotta>

@seaotta

ststimac@microsoft.com

