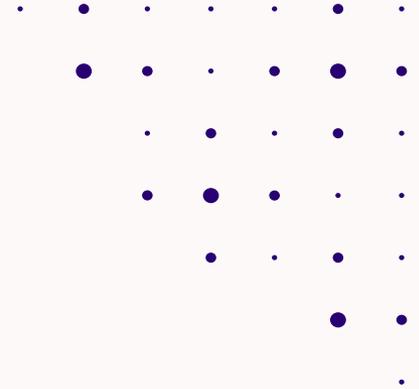


Accessibility Engineering

Lass die Web-Plattform die Arbeit machen

Alexander Lehner (CPACC), 8. Juli 2025



Agenda

1. Was ist Accessibility Engineering?
2. Die grundlegenden Werkzeuge des Accessibility Engineerings
3. Native HTML-Features
4. Ein Blick in die Zukunft
5. Q&A

Was ist Accessibility Engineering?

(Frontend) Web Developer:in



Bauen und warten Websites



Nutzen HTML, CSS, JavaScript etc.



Interagieren mit Backend,
Daten-Layer, etc.



Optimieren Usability, Accessibility,
Ladegeschwindigkeit,
Skalierbarkeit, Sicherheit etc.

Accessibility-Spezialist:in



Wissen über konkrete Bedürfnisse
von Menschen mit Behinderung



Erfahren im Umgang mit assistiven
Technologien



Accessibility Guidelines (WCAG)
und Gesetze (EAA, Section 508)



Technisches Wissen
(Natives HTML, WAI-ARIA etc.)

Was ist Accessibility Engineering?



Accessibility Engineer



Bauen und warten Websites



Nutzen HTML, CSS, JavaScript etc.



Interagieren mit Backend,
Daten-Layer, etc.



Optimieren Usability, Accessibility,
Ladegeschwindigkeit,
Skalierbarkeit, Sicherheit etc.



Wissen über konkrete Bedürfnisse
von Menschen mit Behinderung



Erfahren im Umgang mit assistiven
Technologien

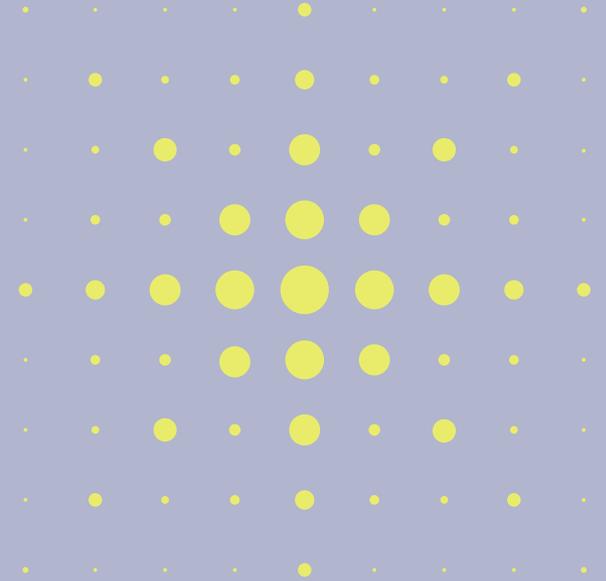


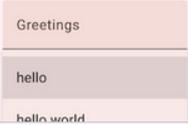
Accessibility Guidelines (WCAG)
und Gesetze (EAA, Section 508)



Technisches Wissen
(Natives HTML, WAI-ARIA etc.)

Die grundlegenden Werkzeuge des Accessibility Engineerings





Autocomplete

Suggests relevant options as the user types.



Badge

A small value indicator that can be overlaid on another object.



Bottom Sheet

A large interactive panel primarily for mobile devices.



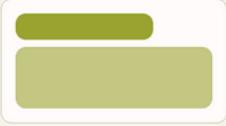
Button

An interactive button with a range of presentation options.



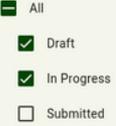
Button toggle

A groupable on/off toggle for enabling and disabling options.



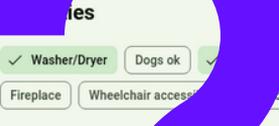
Card

A styled container for pieces of itemized content.



Checkbox

Captures boolean input with an optional indeterminate mode.



Chips

Presents a list of items as a set of small, tactile entities.



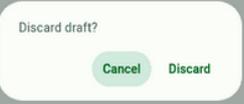
Core

Reusable parts used by other components in the library.



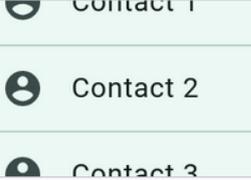
Datepicker

Captures dates, agnostic about their internal representation.



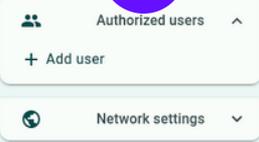
Dialog

A configurable modal that displays dynamic content.



Divider

A vertical or horizontal visual divider.



Expansion Panel

A container which can be expanded to reveal more content.



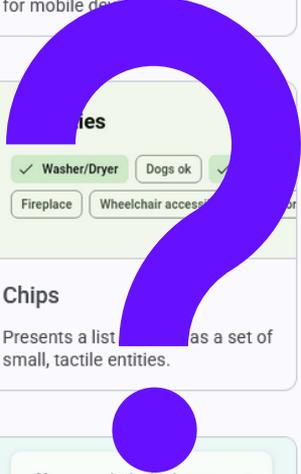
Form field

Wraps input fields so they are displayed consistently.



Grid list

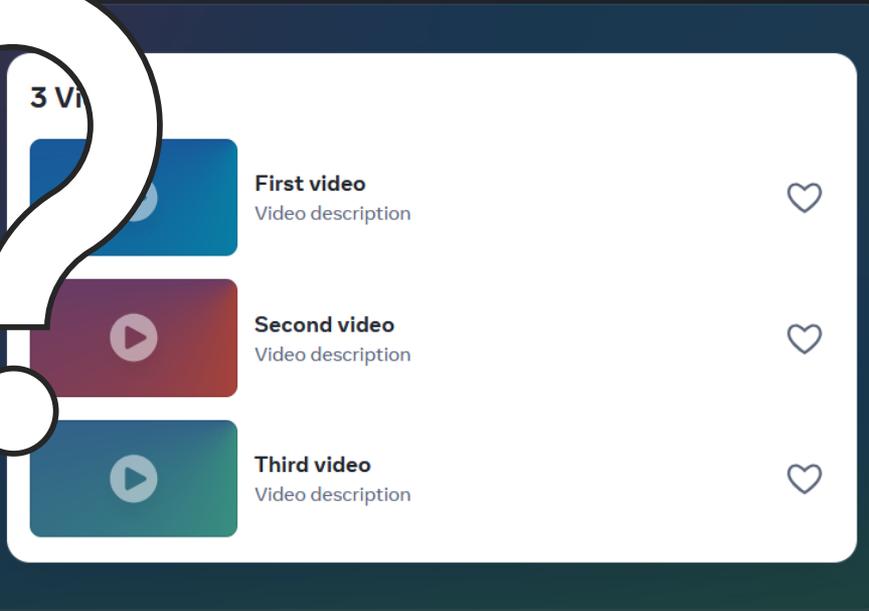
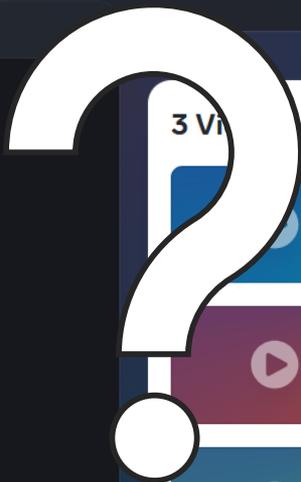
A flexible structure for presenting content items in a grid.





VideoList.js

```
function VideoList({ videos, emptyHeading }) {
  const count = videos.length;
  let heading = emptyHeading;
  if (count > 0) {
    const noun = count > 1 ? 'Videos' : 'Video';
    heading = count + ' ' + noun;
  }
  return (
    <section>
      <h2>{heading}</h2>
      {videos.map(video =>
        <Video key={video.id} video={video} />
      )}
    </section>
  );
}
```

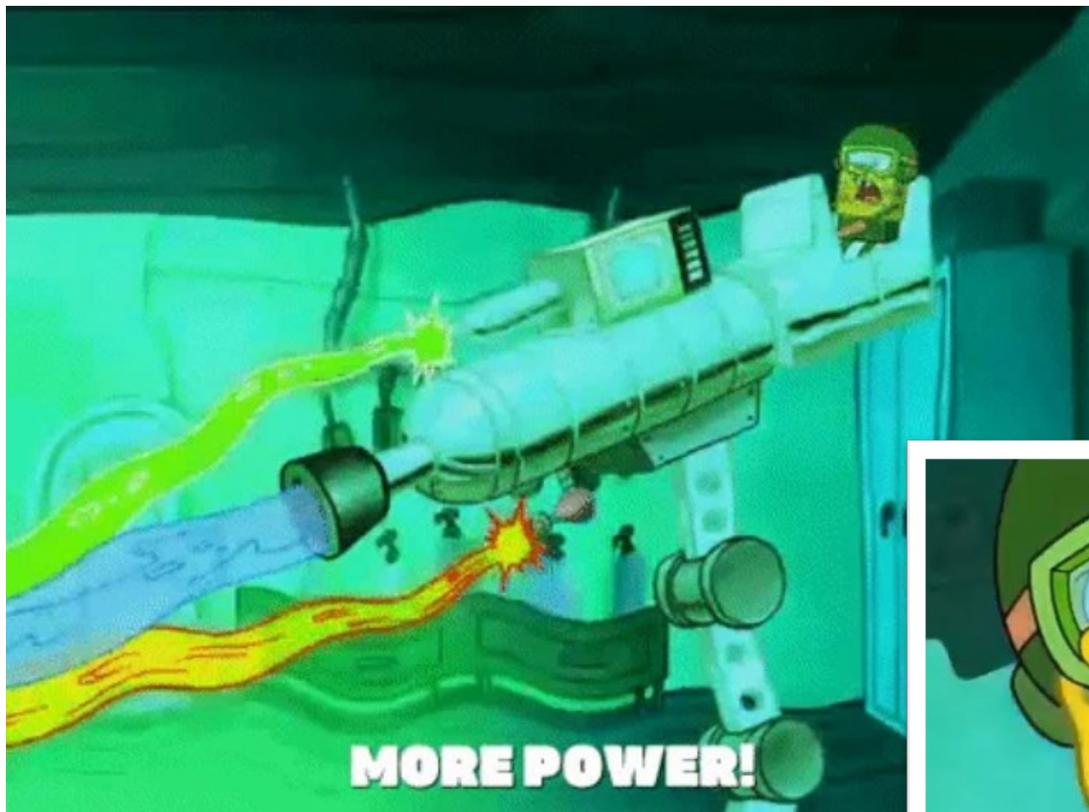


Die Regel der geringsten Macht

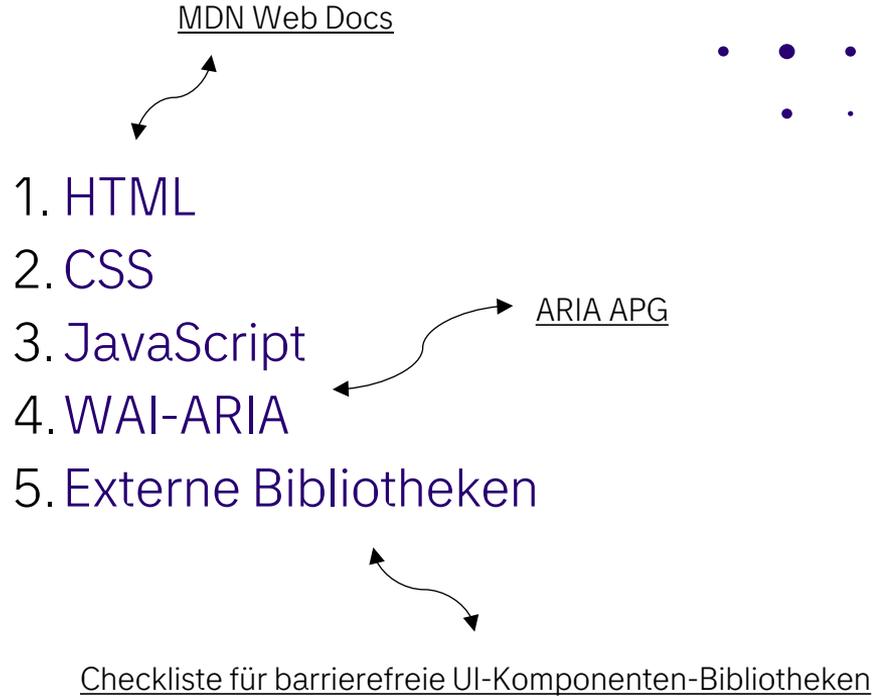


The "Rule of Least Power" suggests choosing the least powerful language suitable for a given purpose.

Quelle: W3C



The Rule of Least Power = HTML First



Baseline 2025

All Baseline 2025 features

14 features

<https://webstatus.dev>

baseline_date:2025-01-01..2025-12-31

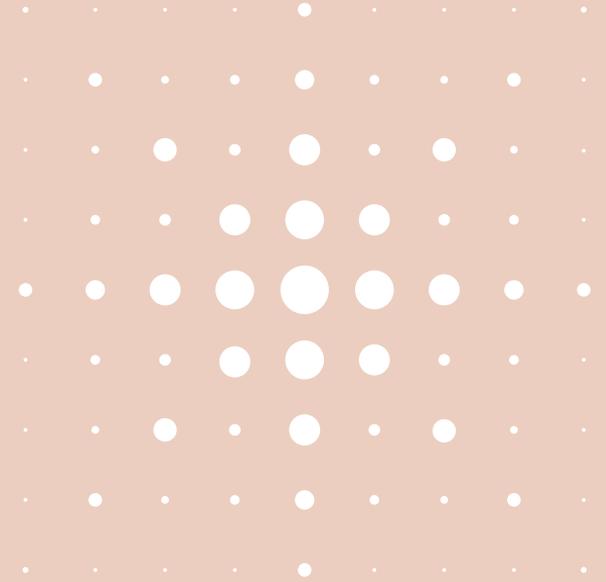
Columns

Export to CSV

Feature	↓ Baseline	Availability				WPT				Usage
		↑↓	↑↓	↑↓	↑↓					
dialog.requestClose()	✓					100%	100%	100%	80.0%	N/A
print-color-adjust	✓					0.0%	0.0%	100%	100%	1.8%
RegExp.escape()	✓					i	i	i	i	0.6%
JSON import attributes <small>INTEROP</small>	✓									N/A
Float16Array	✓					i	i	i	i	1.8%
Single color stop gradients	✓									N/A
Atomics.pause()	✓					i	i	i	i	N/A
ClipboardItem.supports()	✓									0.2%
Iterator methods	✓					i	i	i	i	1.8%
Screen wake lock	✓					94.4%	100%	72.2%	16.7%	1.3%
contenteditable="plaintext-only"	✓					100%	100%	100%	100%	N/A
Intl.DurationFormat	✓					i	i	i	i	<0.1%
Popover <small>TOP HTML</small>	✓					97.0%	98.5%	73.1%	70.1%	58.8%
Promise.try()	✓					i	i	i	i	2.9%

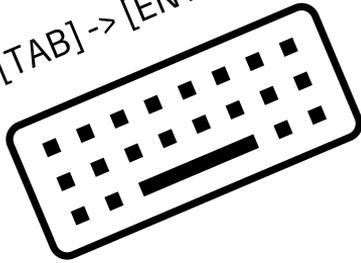
Native HTML Features

(mit breiter Unterstützung)

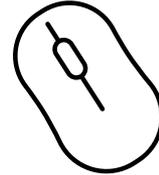


Das <button> Element: Der Allrounder

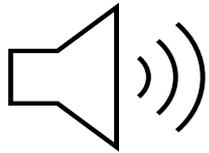
[TAB] -> [ENTER]



((CLICK))



Button



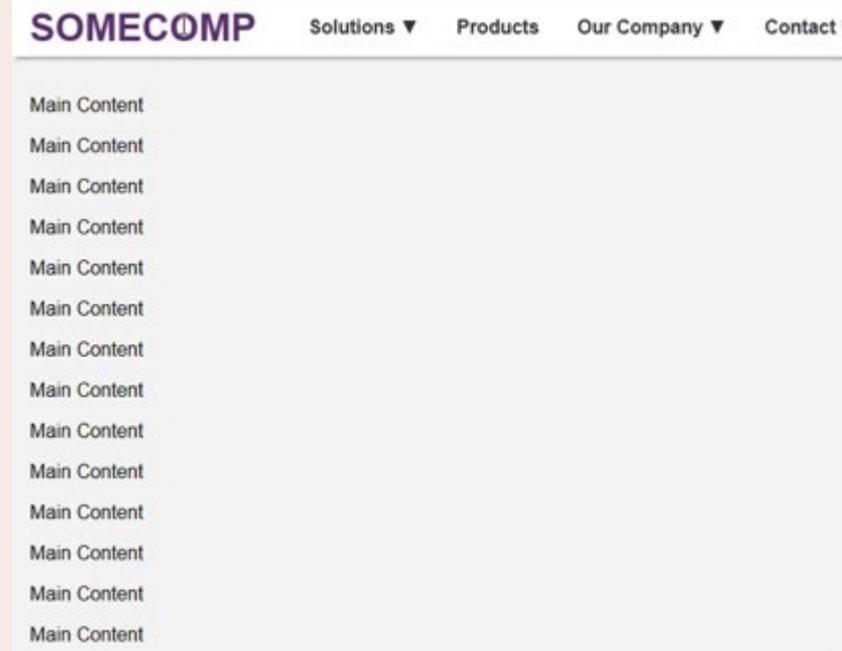
Ich bin ein Button!



“Click” Event Handler

Das <button> Element: Der Allrounder

- **Popover API** (Baseline Newly Available)
 - Button kann verknüpftes Popover öffnen
 - Demo: Navigations-Menü
- **Invoker Commands API** (experimentell)
 - Dem Button ein Verhalten zuweisen
 - Beispiel: Einen Dialog öffnen



Das <dialog> Element: Immer On Top

Demo: ASF
Rastanlagen

A4 Ost Autobahn 8 X

km	Bezeichnung	Erreichbar in Richtung
2,45	Simmering WC 🍴 🚰 🚰 🚰 🚰	Nickelsdorf
22,33	Fischamend WC 🍴 🚰 🚰 🚰 🚰	Nickelsdorf
22,48	Maria Ellend WC 🍴 🚰 🚰 🚰 🚰	Wien
28,04	Göttlesbrunn WC 🍴 🚰 🚰 🚰 🚰	beide Richtungen
46,23	Parndorf WC 🍴 🚰 🚰 🚰 🚰	Wien
47,0	Neusiedl WC 🍴 🚰 🚰 🚰 🚰	Nickelsdorf
64,65	Nickelsdorf Süd WC 🍴 🚰 🚰 🚰 🚰	Nickelsdorf
64,97	Nickelsdorf Nord WC 🍴 🚰 🚰 🚰 🚰 🚰 🚰 🚰	Wien

A4 Ost Autobahn, km 47,0
Erreichbar: Richtung Nickelsdorf

Quelle: ASF IN A | G

Rastplatz Neusiedl

[ROUTE PLANEN](#)

Stellplätze

31 PKW Stellplätze **93** LKW Stellplätze

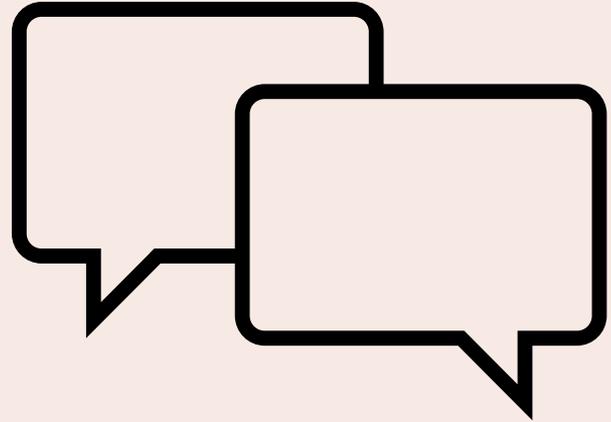
Webcams +

Weitere Rastanlagen Services +

< A4, km 46,23 A4, km 64,65 >

Das <dialog> Element: Immer On Top

- Nativer <dialog> ist **“Baseline Widely Available”**
- **Eingebaute Barrierefreiheit:**
 - Automatischer Tastaturfokus beim Öffnen
 - Hat die semantische Rolle „Dialog“
 - Verdeckter Inhalt ist inaktiv (Benutzer können den Dialog nicht versehentlich verlassen)
 - Schließen des Dialogs mit der ESC-Taste
 - Beim Schließen wird der Fokus zurück zum Auslöser bewegt
- Weitere Infos und Demo



Frei gestaltbare Formular-Elemente

- Native Formular-Elemente sind grundsätzlich für Barrierefreiheit gebaut
- Das Styling der meisten Formularelemente kann beliebig angepasst werden:
 - Eingabefeld: `<input type="text">`
 - Checkboxen: `<input type="checkbox">`
 - Radio Buttons: `<input type="radio">`
 - Auswahlliste: `<select>` (mit Einschränkungen)
- [Demo: Formular mit Custom-Styling](#)
- [Blogartikel mit weiteren Infos](#)

Checkout

Shipping address

First name * Last name *

Address line *

Country

Payment method

Name on card * Card number *

Expiry date * CVV *

Last three digits on signature strip

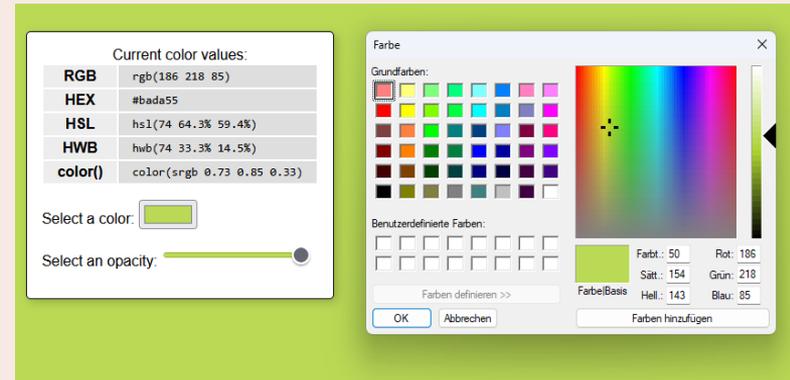
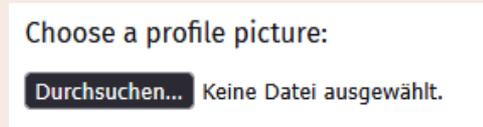
Pay by Card Pay by Transfer

Save the information for later

PURCHASE

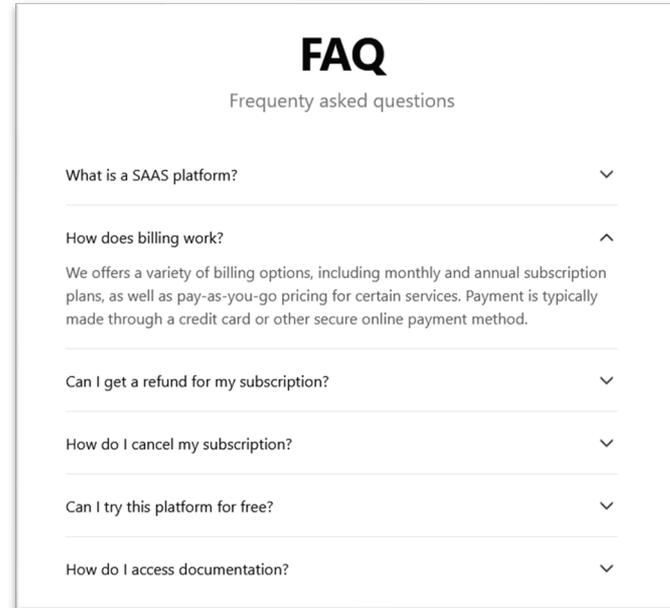
Nicht gestaltbare Formular-Elemente

- Leider können nicht alle native Formular-Elemente frei gestaltet werden:
 - Color Picker
 - Date Picker
 - File Picker
 - Optionen-Liste des `<select>`-Elements
- Coming Soon:
 - Anpassbares `<select>`-Element

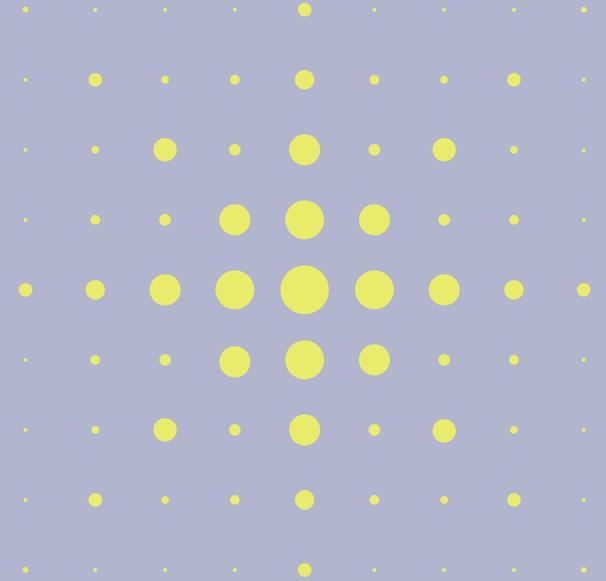


Ausklappbare Bereiche (Akkordeon etc.)

- Natives `<details>` Element erzeugt einen ausklappbaren Bereich (Disclosure Widget)
 - Basis-Feature ist **“Baseline Widely available**
 - Unterstützt seiteninterne Suche des Browsers
 - Nutze **“name”-Attribut**, um mehrere Bereiche zu verbinden und ein Akkordeon-Widget zu erstellen.
 - [Demo: Doctor Who FAQ](#)
- Wird in Zukunft einfacher zu stylen sein:
 - Grid und Flex-Layouts
 - `::details-content` Pseudo-Element
 - Verbesserung sind Teil von [Interop 2025](#)
 - [Chrome: More options for styling <details>](#)

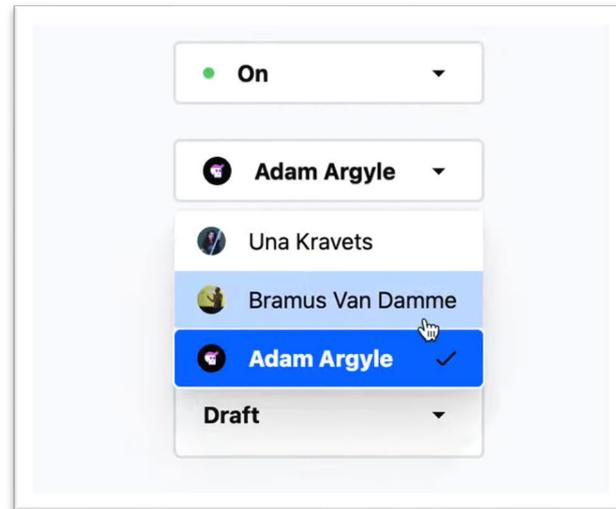


Ein Blick in die Zukunft



Anpassbares <select>-Element

- Opt-in mit „[appearance: base-select](#)“
- Alle Elemente sind frei gestaltbar:
 - Kontroll-Button
 - Liste an Optionen
 - Die ausgewählte Option
- Aktuell nur in [Chrome/Edge 134+](#) unterstützt
- Experimentell: Die [Spezifikation der WHATWG-Community](#) ist noch nicht final
- Demos:
 - [The <select> element can now be customized with CSS](#)
 - [Custom <select> boilerplate + transitions](#)



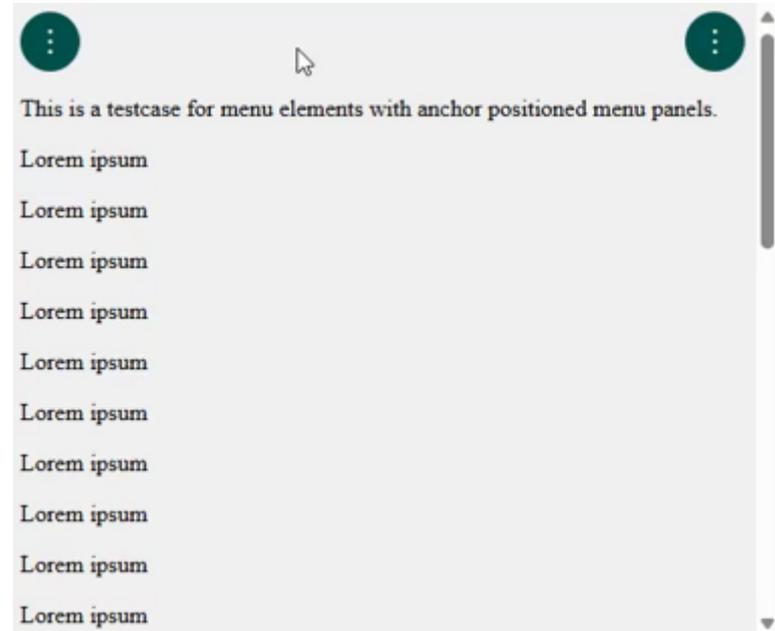
Popover API + CSS Anchor Positioning

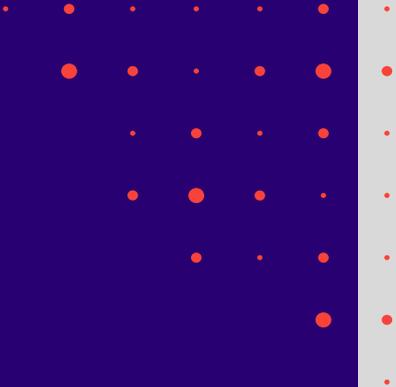
- Popover API:
 - Inhalte oberhalb der restlichen Seite anzeigen
 - Baseline seit Jänner 2025
- CSS Anchor Positioning
 - Inhalte relativ zu anderen Inhalten auf der Seite positionieren bzw. verankern
 - Teil von Interop 2025 (sollte bis Ende 2025 in allen Browsern verfügbar sein)



Use Case: Aktionsmenü

- Kontroll-Button öffnet Menü-Panel:
 - Verknüpfung über Popover API
 - Browser kommuniziert an Screenreader, ob Panel geöffnet ist
 - Korrekte Fokusreihenfolge
 - Schließen mit Esc-Taste
- Positionierung mit CSS Anchor Positioning
- Notwendige manuelle Schritte:
 - ARIA-Rollen „menu“ und „menuitem“
 - Roving Tabindex und Pfeiltasten-Navigation mit Javascript
- Weitere Infos: [Menu and Menubar Pattern \(ARIA APG\)](#)
- [Demo: Menu Anchor Positioning Testcase](#)





Q&A

Blog: <https://oidaisdes.org/>

