



ANGULAR  
**ARCHITECTS**

# WAI ARIA

Alexander Thalhammer | @LX\_T

# WAI ARIA

- Web Accessibility Initiative
- Accessible Rich Internet Applications
- WAI ARIA
  - Roles
  - Attributes

# WAI ARIA

- Adds HTML-attributes to elements
  - improving accessibility for dynamic web content
  - especially for people with disabilities
- The attributes help assistive technologies
  - like screen readers, Braille, ...
  - understand and navigate web content more effectively
- Defining roles, states, properties
  - relationships of elements
  - allowing Devs to create accessible & inclusive web apps

# NG ARIA [attr]

- ARIA attributes enable modifying
  - an element's states and properties as defined in the A11y tree
  - <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Attributes>
- When binding to ARIA attributes in Angular, you must use the "attr." prefix

```
<!-- Use attr. when binding to an ARIA attribute -->  
<button [attr.aria-label]="myActionLabel">...</button>
```

```
<!-- Static ARIA attributes require no extra -->  
<button aria-label="Close">...</button>
```

# ARIA vs. Semantic HTML

- Semantic HTML
  - preferred
  - native elements like `<button>`, `<label>`, `<nav>` and `<header>`
  - provide built-in accessibility features
- ARIA
  - is our last resort
  - when semantic HTML not possible / not applicable



# WAI-ARIA Essentials

## – Roles

- enhance a11y by providing semantic information, when native HTML elements cannot be used
- e.g. `<div role="button" tabindex="0" (click)="onClick()">Click Me</div>`

## – Attributes

- enhance by providing additional semantic information
- states (dynamic / temporal) vs. properties (permanent)

# ARIA Roles

- define what an element is or does
  - role="button" (prefer <button>)
  - role="navigation" (prefer <nav>)
  - role="complementary" (prefer <aside>)
  - role="main" (prefer <main>)
  - role="form" (prefer <form>)
  - role="banner"
  - role="search"
  - role="tablist" & role="tabpanel"
  - role="presentation"
    - removes default behaviour e.g. <button>

# ARIA Roles – Example

```
<app-dialog role="dialog" aria-labelledby="dialogTitle" aria-modal="true">  
  <h2 id="dialogTitle">{{ dialogTitle() }}</h2>  
  <p>{{ dialogContent() }}</p>  
  <button type="button" (click)="onClose()">Close</button>  
</app-dialog>
```



# ARIA Roles – Full List

– <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles>

# ARIA Attributes I

- ARIA attributes enable modifying an element's states and properties as defined in the A11y tree
- Widget attributes
  - aria-disabled (state)
  - aria-required (property)
  - aria-expanded (state)
  - aria-hidden (state)
  - aria-invalid (state)
- Live region attributes (states)
  - aria-live ('assertive' → error, 'polite' → info)
  - aria-busy (loading spinner)

# ARIA Attributes II

- Drag-and-Drop attributes
  - aria-grabbed (state)
- Relationship attributes (properties)
  - aria-label
  - aria-labelledby
  - aria-describedby
  - aria-controls

# ARIA Attribute – Example

```
<button  
  type="button"  
  aria-label="Toggle navigation menu"  
  [attr.aria-expanded]="isMenuOpen() ? 'true' : 'false'"  
  (click)="onToggleNav()">  
  <i class="icon-menu"></i>  
</button>
```

# ARIA Attributes – Full List

– <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Attributes>





Questions?