

# Full Stack Application Development

**Accessibility**

**Akshaya Ganesan**



# UX based on Technologies used

- Consider the technologies that people use to experience our websites
  - devices,
  - browsers,
  - operating systems, and
  - assistive technologies
- How to design for them
- **Responsive design** is about creating fluid designs through HTML and CSS that adapt seamlessly to screen size, resolution, and aspect ratio, so that the layout remains optimally usable and attractive
- **Accessible design** is about creating designs that are usable and enjoyable by people with disabilities, including physical, sensory, cognitive, and neurological problems
- **Universal design** is about creating designs that are usable and enjoyable by everyone, regardless of age, status, culture, ethnicity, or ability

# Accessibility

- Consider the technologies that people use to experience our websites
  - devices,
  - browsers,
  - operating systems, and
  - assistive technologies
- Accessible design is about creating designs that are usable and enjoyable for people with disabilities.

# Designing for accessibility

- The most important way to design for assistive technologies is to design with different users in mind, following usability guidelines
- Designing for keyboard input
- Designing for Screen Readers
- Careful color and color contrast choices
- Web accessibility is about removing barriers that prevent people with disabilities from accessing websites.

# Principles of accessibility

- The **Web Content Accessibility Guidelines (WCAG 2.0)** by W3C defines
- Four principles of accessibility.
- **Perceivable:** Information and other interface elements must be visible to everyone.
- **Operable:** Everyone must be able to navigate around any website.
- **Understandable:** The information on websites must be easily understandable.
- **Robust:** The content and its interpretation must be reliable, and compatible with different devices, browsers, and assistive technologies.

# Guidelines

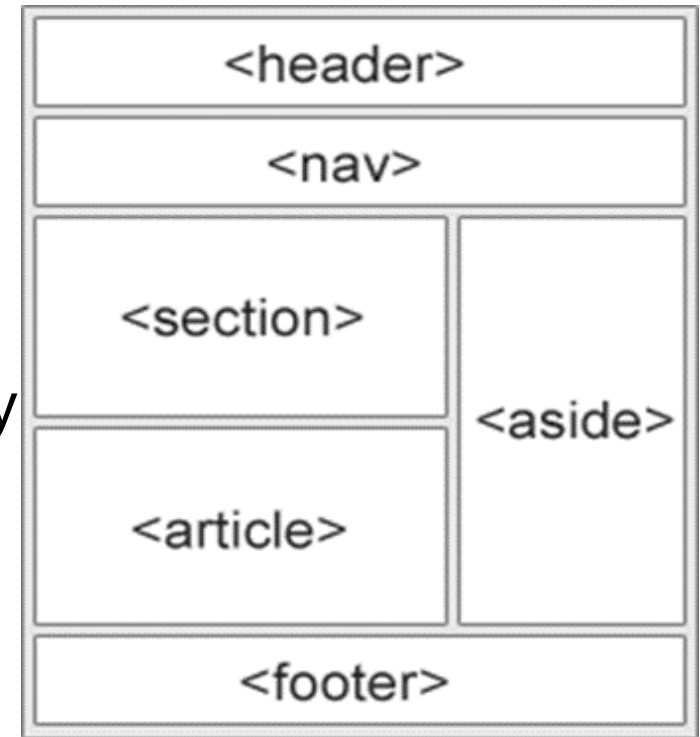
- Using semantic HTML elements.
- Use Text Alternatives for non text content like images
  - `<img src="" alt="Logo" />`
- Ensure that form elements are accessible.
- Arrange the HTML document with a logical and hierarchical structure.
- Use appropriate heading elements
- Ensure that all interactive elements are accessible via the keyboard.
- Maintain sufficient contrast between text and background colors to enhance readability.

# An army of <div> elements

```
<div class="container">
  <div class="row ">
    <div class="col-2 bg-primary">first column</div>
    <div class="col-6 bg-secondary">second column</div>
    <div class="col-4 bg-primary">third column</div>
  </div>
</div>
<hr/>
<div class="container">
  <div class="row ">
    <div class="col-sm-2 bg-primary">first column</div>
    <div class="col-sm-6 bg-secondary">second column</div>
    <div class="col-sm-4 bg-primary">third column</div>
  </div>
<hr/>
<div class="row ">
  <div class="col-md-2 bg-primary">first column</div>
  <div class="col-md-6 bg-secondary">second column</div>
  <div class="col-md-4 bg-primary">third column</div>
</div>
```

# Semantic Elements

- header: `<header>`.
- navigation bar: `<nav>`.
- main content: `<main>`,
- Main content can be organized into subsections by
  - `<article>`, and
  - `<section>`
- sidebar: `<aside>`; often placed inside `<main>`.
- footer: `<footer>`.





# ARIA



- The Web Accessibility Initiative's Accessible Rich Internet Applications specification (WAI-ARIA)
- ARIA provides additional information to assistive technologies about the roles, properties and states of elements.

# Example

```
<button aria-expanded="false" onclick="toggleCollapsible()">
```

Toggle Section

```
</button>
```

```
<div class="content" aria-hidden="true">
```

```
  <p>This is the content of the collapsible section.</p>
```

```
</div>
```

# Testing for accessibility

- **Keyboard only:** Check if you can access all parts of your website while only using the keyboard
- **Screen reader:** Download a screen reader or use the built in one for your OS and view your website through it.
- **Magnification:** Enlarge your font to 200%.
- **Checklists:** There are many accessibility checklists available to check your designs and HTML against, based on the WCAG guidelines.
  - WebAIM (<http://webaim.org/standards/wcag/checklist/>)
- **WAVE:** WebAIM also provides the WAVE extension for Chrome and Firefox, which checks accessibility automatically

# Self Reading



- <https://testsigma.com/tools/accessibility-testing-tools/>

# Thank You!

