# Inclusive Design

* describe how involving users with disabilities early and throughout in the design and development process leads to better accessibility of websites and applications
* explain strategies and assistive technologies that people with disabilities use to interact with the Web
* research and communicate applicable design requirements, such as those defined by accessibility standards and policies
* manage the involvement of people with disabilities in different stages of the design and development process, for example in focus groups, personas, scenarios, and usability testing
* ensure representation of diverse groups of users with disabilities, including people with various levels of skills using assistive technologies and adaptive strategies
* ensure people with disabilities can provide informed consent, respond to questions, and otherwise participate in the design and development processes in an accessible way

# Page Layout

* explain how people with disabilities identify and distinguish user interface components based on different features such as color, structure, layout, pattern, spacing, position, and naming
* create designs with sufficient contrast ratios between foreground user interface components, including text, images of text, and controls, and their corresponding background
* ensure information is not conveyed through color alone without supplemental text cues, patterns, and icons
* create designs with clear and distinguishable focus indicators, for example using borders, background color, and highlighting
* ensure different page regions and user interface components are placed where users expect them and are distinguishable from one another, for example through through color, typography, patterns, spacing, position and naming
* provide clear and descriptive heading texts that allow users to identify different sections of the web page and parts of the content
* identify related requirements for developers to ensure that headings are coded appropriately and are nested according to the hierarchical structure of the page

# Navigation and Orientation

* explain strategies that people with disabilities use to navigate and orient themselves in web pages and applications
* create designs that allow users to bypass blocks of repeated content, such as navigation links, header contents, or advertisements
* create designs that allow for multiple navigational mechanisms, such as menus, search boxes, breadcrumb trails, tables of contents, and sitemaps
* create designs that allow users to stop, adjust, and extend existing time limits set by the content
* identify related requirements for developers to ensure that navigational mechanisms can be used with the keyboard and other input devices

# Information Design

* explain how people with disabilities perceive and access different types of information and perform different tasks within websites and applications
* explain the use of headings to divide longer passages of text into smaller pieces
* identify different uses of text, for example for labels, static content, and dynamic messages, and describe alternative ways to convey information presented in text, such as using graphics, color, or shapes
* identify and define different types of tables from an accessibility perspective, such as layout, simple, and complex tables
* describe the accessibility challenges that complex tables create, and explain different solutions including splitting complex tables into simpler tables, lists, and plain text where possible
* ensure different journeys, flows, and processes are accessible to all users, for example different flows to perform a task within a website or different ways to check the details of a given product
* describe different techniques to group information in form fields to make it more understandable by all users, such as splitting the form into several more comprehensive steps and providing clear instructions for users to understand what is expected from their input
* identify related requirements for developers to ensure that instructions for form controls are programmatically associated to their corresponding control and that table header cells are programmatically associated to their corresponding data cells

# Images and Graphics

* explain how people with disabilities obtain information contained in images through different methods, such as through text alternatives and descriptions
* explain how images help people with different types of disabilities better understand the information, context, and functionality
* identify and describe different types of images from an accessibility perspective, such as informative, textual, decorative, functional, and complex images
* provide appropriate text alternatives for images included in the design phase, such as logos or graphical components
* provide text that allows users to identify graphical interactive elements such as links or menu items
* describe alternative ways to convey information presented in text, such as using graphics, color, or shapes
* create designs that consider descriptions for images [and graphics] such as charts or diagrams
* identify related requirements for developers to ensure alternatives and descriptions are programmatically associated to their corresponding images and graphics
* identify related requirements for content authors to provide appropriate text alternatives for different types of images, including informative, textual, decorative, functional, and complex

# Multimedia and Animations

* identify accessibility requirements for different types of audio or video content
* create designs that consider alternatives to multimedia content, such as transcripts for audio and audio described content,
* create designs with mechanisms to pause, stop, and hide any moving, blinking, scrolling, auto-updating, and auto-playing content, such as animations, carousels, or auto-playing audio
* identify related requirements for developers to ensure alternatives and descriptions are programmatically associated to their corresponding media content
* identify related requirements for content authors to provide appropriate text alternatives for different types of media content, including video and audio

# Interaction and Feedback

* explain strategies that people with disabilities use to identify, distinguish, and operate interactive user interface components, such as forms and custom widgets
* ensure all interactive user interface components can be operated using the keyboard
* provide clear and distinguishable names and instructions that allow to identify interactive user interface components, such as form fields and custom widgets
* provide meaningful and descriptive notifications about imminent time limits, status changes, and feedback from user input, including errors, suggestions for correction, and success messaged
* create designs with meaningful sequence of interactive user interface components
* create designs with clear focus indicators both individually and when operated in the context of a rich application
* define custom interaction patterns, such as where to place focus when a dialog is open and closed, and what the focus order should be within those dialogs
* define mechanisms that allow users to obtain information about custom interaction patterns, such as specific keyboard shortcuts for applications and expected behavior of custom widgets
* identify related requirements for developers to use native HTML elements to the extent possible and describe their benefits in relation to non-native elements
* identify related requirements for developers to ensure that properties of custom user interface components, such as names, states, and instructions are perceived visually and can be programmatically determined
* identify related requirements for developers to write code for specific keyboard interactions that custom widgets may require

# Responsive Design

* explain how people with disabilities use different screen sizes, screen configurations, and style sheets to access websites and applications
* create designs with target sizes and spaces to support tapping by people with mobility impairments and people using different input methods
* ensure meaningful sequence, content, and functionality are preserved when using different viewport sizes, screen configurations, and devices to access content
* create designs that preserve content and functionality when line height as well as spacing between paragraphs, words, and letters are modified by the user
* create designs that support text resizing without loss of content and functionality
* create designs that do not restrict the content view and operation to a single display orientation, such as portrait or landscape
* ensure input modalities do not rely solely on multipoint or path-based gestures, such as swipe or pinch
* ensure that mechanisms to undo or abort an action carried out with pointer operations are always available