



# Improving Accessibility Through Process, Style Guide and Pattern Library

*March 21, 2018*

# Authors

## Mark Stimson, Ph.D.

Digital Accessibility Manager  
Digital Experience Center  
[Mark.X.Stimson@kp.org](mailto:Mark.X.Stimson@kp.org)



## Vita Zavoli

Lead Digital Accessibility Specialist  
Digital Experience Center  
[Vita.Zavoli@kp.org](mailto:Vita.Zavoli@kp.org)

## Eric Yeh

Front End Developer  
Digital Experience Center  
[Eric.B.Yeh@kp.org](mailto:Eric.B.Yeh@kp.org)

## Nancy Crandall

Front End Developer  
Digital Experience Center  
[Nancy.S.Crandall@kp.org](mailto:Nancy.S.Crandall@kp.org)

# Speakers

## Mark Stimson, Ph.D.

Digital Accessibility Manager

Kaiser Permanente

Digital Experience Center

[Mark.X.Stimson@kp.org](mailto:Mark.X.Stimson@kp.org)



## Vita Zavoli

Lead Digital Accessibility Specialist

Kaiser Permanente

Digital Experience Center

[Vita.Zavoli@kp.org](mailto:Vita.Zavoli@kp.org)



## Nancy Crandall

Front End Developer

Digital Experience Center

[Nancy.S.Crandall@kp.org](mailto:Nancy.S.Crandall@kp.org)



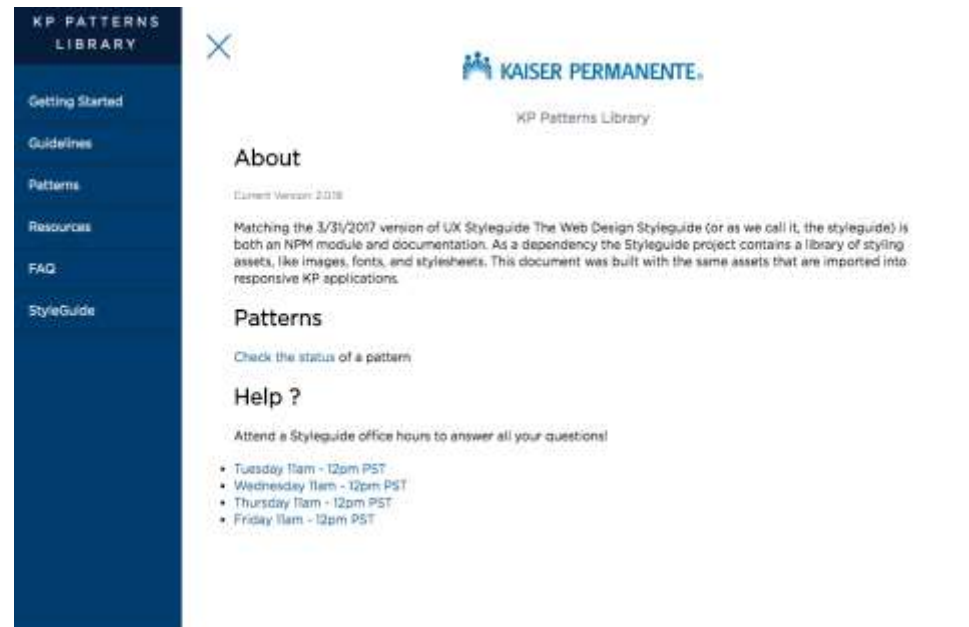
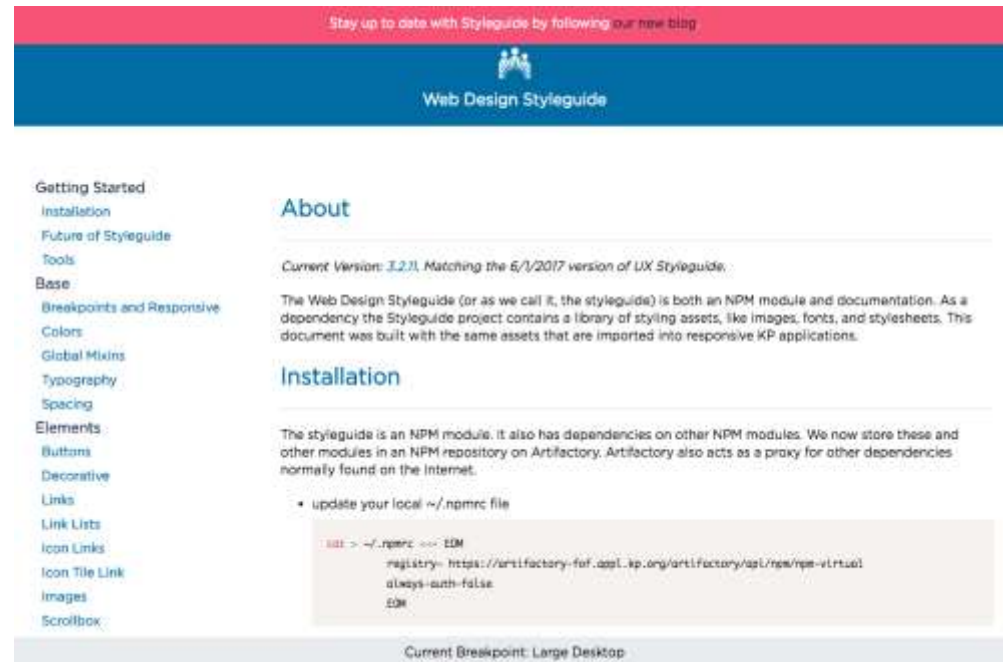
# Agenda

- Objective
- Introduction
- Background on Kaiser Permanente
- Style Guide and Pattern Library Summary
- Accessibility Process
- Examples



# Objective

Shared lessons learned from evolving accessibility process with Style Guide Designers and Developers



# Introduction

Kaiser Permanente has attempted to solve the problem of demand on accessibility testing and development with an accessible Style Guide and Pattern Library which fits into the Adobe Experience Manager (AEM) framework, and a process for ensuring updates and changes are accessible.

KP Patterns Library > Carousel

## Carousel

**Demo** Installation Accessibility Markup JS/jQuery Style

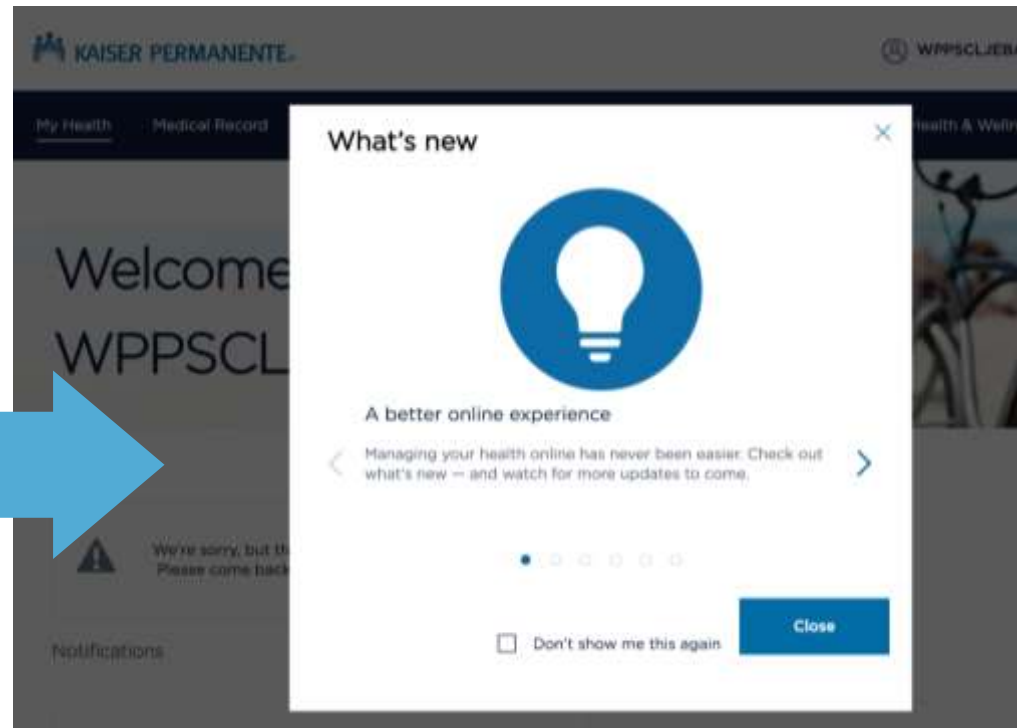
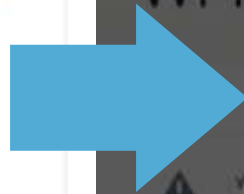
### Demo

The carousel pattern, as examples below, are intended to be customized to your projects specifications. These are not out of the box solutions. If you would like to add additional styles to the project feel free to create a Pull Request and contribute to the [carousel pattern](#)

600x300

### Title 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam id ex vel lacus euismod tempus. Ut vulputate pellentesque magna, vitae bibendum enim blandit in.



# Our Story

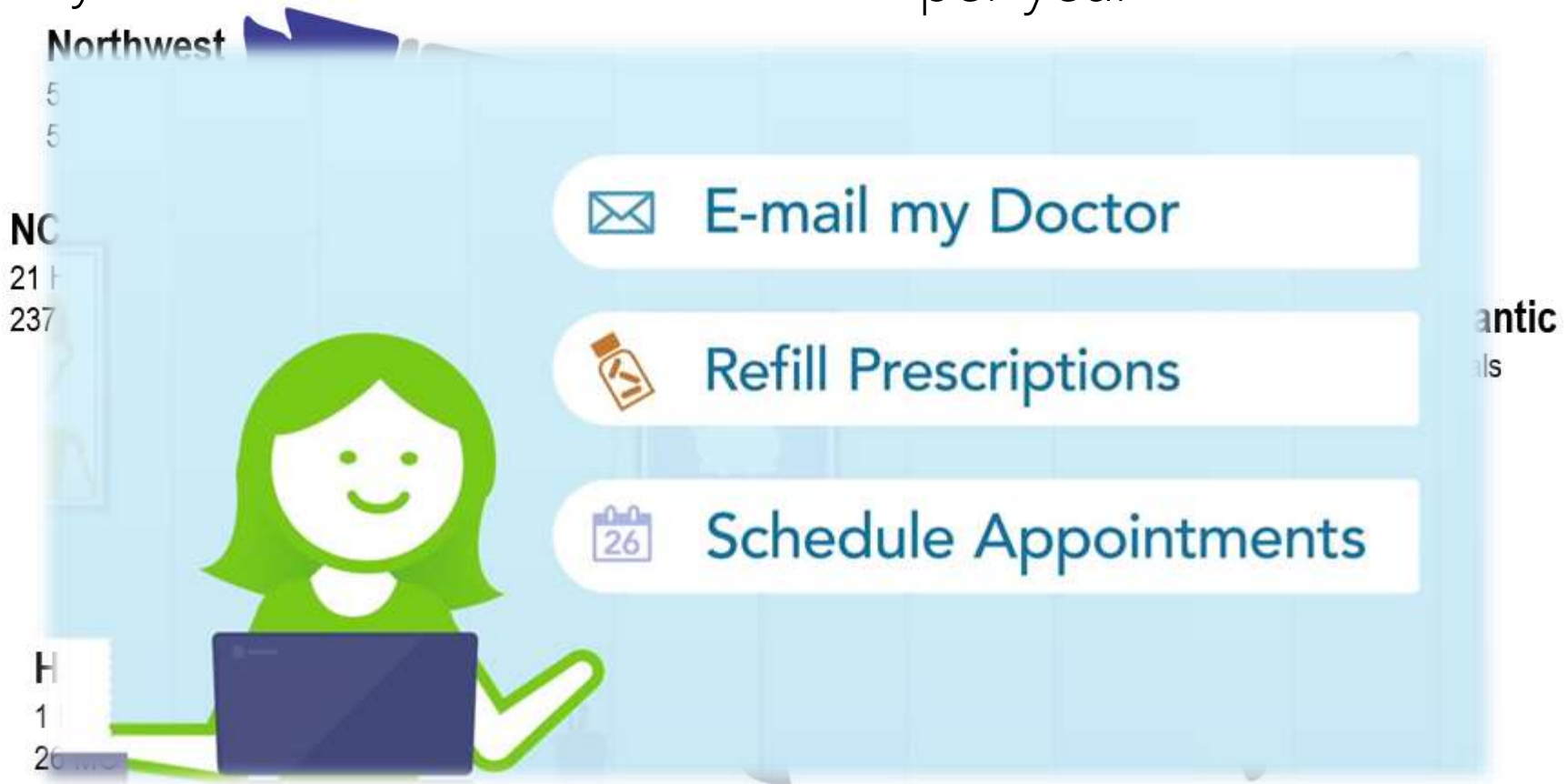


Kaiser Permanente's Long Journey to Success



# Kaiser Permanente is...

- Largest **care & coverage** health care organization  
Non-profit offering lowest cost with best care
- 12 million members in 8 states • 82 Hospitals
- Providing health care for more than 300 million website visits  
75 years per year





# KP's Digital Experience...

## History

- Began in 1996
- Integrated into first large-scale EMR
- Care integrated digitally (appointments, visits, Rx, labs, records, preventions, etc.)

## Now

- 70% of members now registered on [kp.org](https://kp.org)
- Half-million visits/day
- Majority from mobile

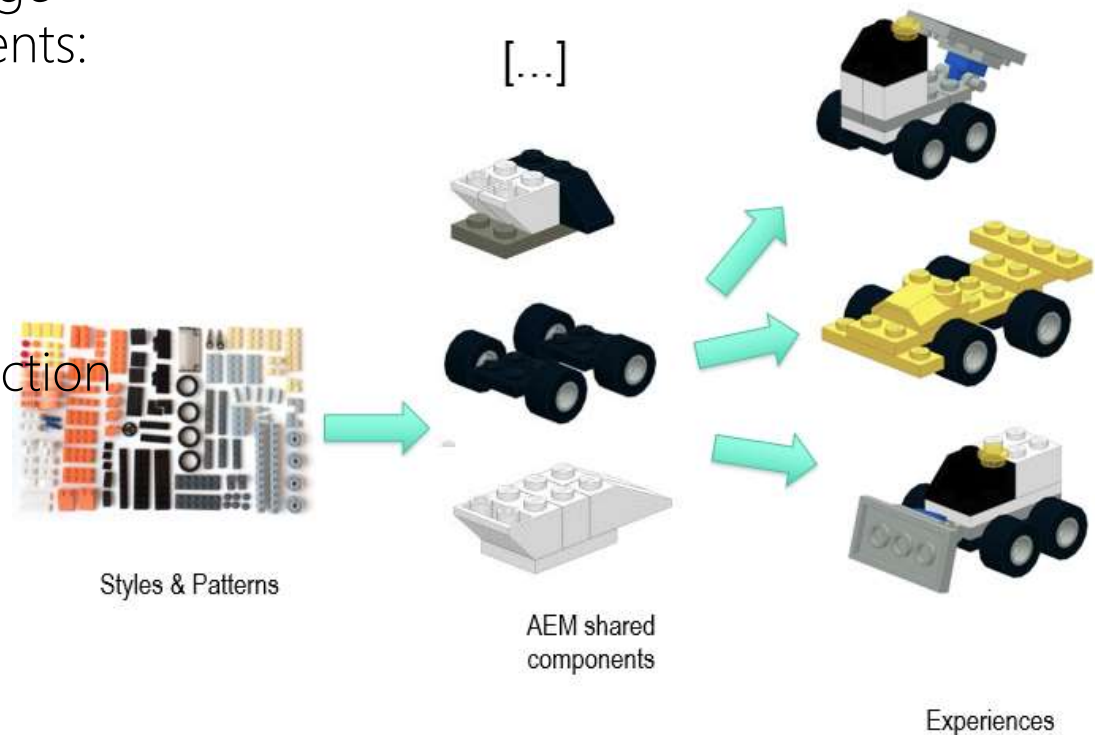
# KP's Digital Inclusion...



# KP Style Guide and Pattern Library Summary

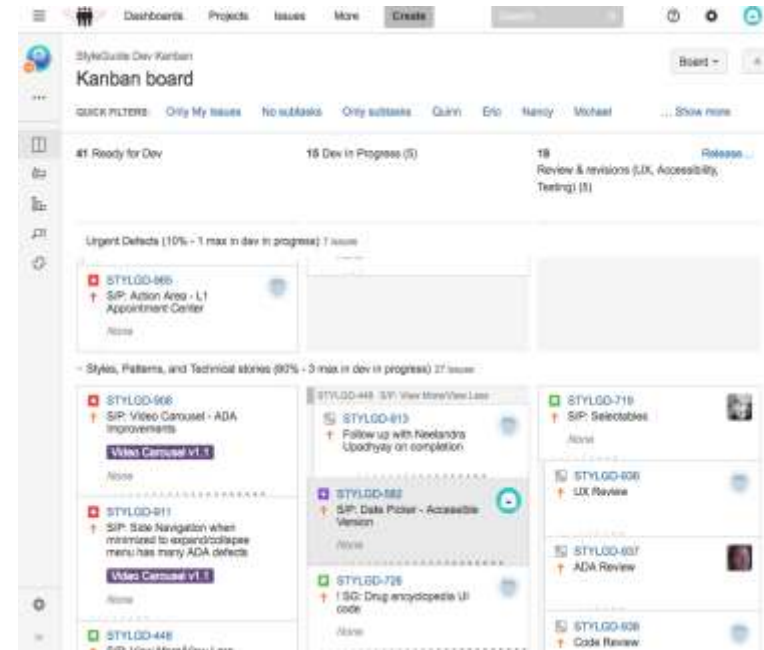
# About Style Guide and Pattern's library

- The Style Guide and Pattern Library are building blocks
- Style Guide is a NPM package HTML/CSS for simple elements:
  - buttons
  - colors
  - headings, etc..
- The Pattern library is a collection of complex patterns as standalone plugins:
  - Accordions
  - carousels
  - modals etc



# Style Guide and Pattern library process

- The Style Guide is developed using an Agile process.
  1. The design team provides the screen markup/composition to the development team.
  2. FED then creates a story and determines how many hours that pattern will take to build. It includes information needed for making the component accessible.
  3. The story is given a sizing number to indicate the complexity of the task, then placed into JIRA to manage the workflow.



# The Accessibility Process

# Accessibility process

- Accessibility team meets with UX design once a week to review comps. Comps are distributed to the Accessibility team before the meeting occurs. At least two team members attend these meetings.
- Style Guide / Pattern library grooming session
- Accessibility team meets with FED for a working session once a week or as needed. These meetings also have several Accessibility team members present depending on specialty.
- The component then goes through a testing process using a testing grid. The aim is to make sure that no high-end severity defects are present.
- Accessibility SMEs across departments meet to review defects to suggest changes for implementation such as ARIA code, role values, keyboard and screen reader fixes, etc.
- Accessibility and FED meet and include others from different departments as needed once a week to discuss corner cases and strategize regarding website global issues that are being considered for the future.



# Testing on a LARGE Scale

- Over: 90 combinations across 17 assistive technologies

Browsers	Keyboard	JAWS	NVDA	Voice Over	Chrome Dev Tool	Color Contrast	Visual ARIA	AMP	Browser Magnify (200%)	Talkback	Android Magnifier	iOS Zoom	iOS Assistive Touch	iOS Inverse Colors	iOS Switch Control	Dragon	High Contrast Mode	JAWS 2018	MAGic	ZoomText	System Access	Window Eyes	Narrator	Win On-Screen Keyboard
Chrome (Win Desktop)	TEST	TEST	TEST	N/A	TEST	TEST	TEST	TEST	TEST	N/A	N/A	N/A	N/A	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope
Safari (iPhone)	Out of Scope	N/A	N/A	TEST	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	N/A	N/A	N/A	N/A	N/A	N/A	Out of Scope	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Internet Explorer	TEST	TEST	Out of Scope	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	TEST	N/A	N/A	N/A	N/A	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope
Firefox	TEST	Out of Scope	TEST	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	N/A	N/A	N/A	N/A	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope
Safari (Mac OSX)	Out of Scope	N/A	N/A	Out of Scope	N/A	Out of Scope	N/A	Out of Scope	Out of Scope	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	N/A	N/A	N/A	Out of Scope	N/A	N/A	N/A	N/A
Chrome (Android)	Out of Scope	N/A	N/A	N/A	N/A	Out of Scope	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	N/A	N/A	N/A	N/A	Out of Scope	Out of Scope	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Edge	Out of Scope	Out of Scope	Out of Scope	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	N/A	N/A	N/A	N/A	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope	Out of Scope

7 Browsers

17 Assistive Technologies

91 Possible Test/Use Cases

14 (15%) Planned Test/Use Cases

# Testing on a Realistic Scale

- 14 combinations across 5 assistive technologies

## Text Area

[styleguide-location/forms.html](#)

[jira task number](#)

TEST GRID									
Browsers	Keyboard	JAWS	NVDA	Voice Over	Chrome Dev Tool	ARIA Code Review	AMP	Browser Magnify (200%)	Color Contrast
Chrome (Win Desktop)	PASSED	#2	PASSED	N/A			#1	PASSED	PASSED
Safari (iPhone)	Out of Scope	N/A	N/A	PASSED	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope
Internet Explorer	PASSED	#2	Out of Scope	N/A	N/A	Out of Scope	Out of Scope	PASSED	Out of Scope
Firefox	PASSED	Out of Scope	#2	N/A	N/A	Out of Scope	Out of Scope	Out of Scope	Out of Scope
PERSON RESPONSIBLE	Tester (WIN7)	Tester (WIN7, WIN10)	Tester (WIN7, WIN10)	Tester (IOS11)	Tester (WIN 10)	Tester(WIN 10)	Tester (WIN7)	Tester (WIN7)	Tester (WIN7)

LEGEND
Defect
Passed
Inconclusive

## SUMMARY OF WCAG DEFECTS

Summary	Severity	Details	Recommendation	Comments
#1) Labels are not associated with <textarea />	S1	<p>These textarea elements do not have a mechanism that allows an accessible name value to be calculated</p> <p>1. &lt;textarea placeholder="Please explain." name="field2" maxlength="255" id="field2" data-show-lines="5" class="text-area_word-count -inky -book" aria-describedby="textarea_count2"&gt;&lt;/textarea&gt;</p> <p>BP: Provide a valid label for form fields</p> <p>AMP Report: <a href="#">http://ampreortlocation</a></p>	<p>When on-screen label is available, use &lt;label /&gt;. The value of &lt;label /&gt; is then explicitly associated with &lt;textarea /&gt; using the "for" and "id" attributes.</p> <p>When on-screen label is not available, you can use the following option to provide the label:</p> <ol style="list-style-type: none"><li>"title" attribute</li><li>"aria-label" attribute</li><li>"aria-labelledby" attribute</li></ol>	<p>The "placeholder" attribute does represent a label or hint. However when it is used, the label or hint may not be detected by assistive technologies. Therefore, an offscreen text for &lt;label /&gt; or a title attribute should be used instead.</p> <p>Also, users will not have the text label available for reference at the same time as filling in a field. This is a problem for user with memory impairments.</p>
#1) Aria-live region read twice.	S3	<p>JAWS with Chrome</p> <p>Example 1</p> <p>Actual Behavior: "Please explain. Edit, use JAWS key + Alt + R to read descriptive text, 255 of 255 characters left, type in text, 255 of 255 characters left" (defect, reads aria-live region twice)</p> <p>JAWS with Internet Explorer</p> <p>Example 1</p> <p>Actual Behavior: "tab panel, Please explain. Edit, use JAWS key + Alt + R to read descriptive text, 255 of 255 characters left, type in text, 255 of 255 characters left" (defect, reads aria-live region twice)</p> <p>NVDA with Firefox</p>		

# Testing on a Realistic Scale (continued)

- 14 combinations across 5 assistive technologies

## Text Area

[styleguide-location/forms.html](#)

[jira task number](#)

DEFECTS BY TEST METHOD (details for accessibility team)			
Reviewer: Tester 1			
Browsers	JAWS (ver 18.0.4534) Issues	Severity	Recommendation
Chrome 64.0.3282.186 (64-bit) (WIN7 and WIN10 Desktop)	Example 1 Actual Behavior: "Please explain. Edit, use JAWS key + Alt + R to read descriptive text, 255 of 255 characters left, type in text, 255 of 255 characters left" (defect, reads aria-live region twice) Example 2 Same, but with 590 characters.	S3	
Internet Explorer 11.0.9600.18920CO (WIN7 and WIN10 Desktop)	Example 1 Actual Behavior: "tab panel, Please explain. Edit, use JAWS key + Alt + R to read descriptive text, 255 of 255 characters left, type in text, 255 of 255 characters left" (defect, reads aria-live region twice) Example 2 Same, but with 590 characters.	S3	

Reviewer: Tester 1			
Browsers	NVDA (ver 2018.1) Issues	Severity	Recommendation
Chrome 64.0.3282.186 (64-bit) (WIN7 Desktop)	Working as expected	N/A	
Firefox 58.0.2 (64-bit) (WIN10 Desktop)	Example 1 Actual Behavior: "255 of 255 characters left, Please explain. edit multi line, 255 of 255 characters left" (defect, reads aria-live region twice) Example 2 Same, but with 590 characters.	S3	

Tester 1 (iphone 6)			
Browsers	Voice Over Issues	Severity / Priority	Recommendation
Safari (iPhone 6, iOS 11.2)	Working as expected	N/A	

# ACCESSIBLE CODE

# Reasons for accessible code and documentation

Issue	Solution
Consuming team developers Style Guide/pattern code have limited accessibility experience	<ul style="list-style-type: none"><li>• Accessible HTML markup examples</li><li>• Offer accessible JavaScript/NPM plugins and templates that meet Accessibility requirements</li><li>• And clear documentation</li></ul>

# Solving Common Accessibility Implementation Issues

Issue	Solution
Copy and Pasting code	<ul style="list-style-type: none"><li>• Offer Reusable code</li><li>• NPM registry to have packages of code</li><li>• Clear documentation</li><li>• Communication</li><li>• Office hours and pair programming sessions to help teams with implementation</li></ul>

# Examples



# What is accessible code?

- Aria-label
- Aria-labelledby
- Aria-describedby
- Aria hidden
- tablist
- Role
- And so much more!

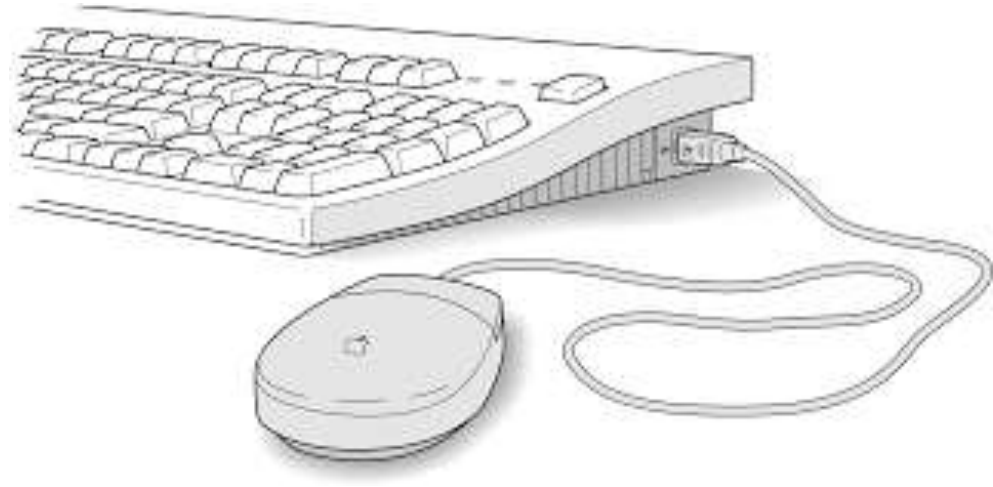
✗ `<nav role='navigation' > learn more </nav>`

○ `<ul role='navigation'>learn more </ul >`

○ `<nav class='side_navigation'> learn more </nav>`

# Accessible JS Events

- Mouse Dependent
  - onMouseOver,
  - onMouseOut
- Keyboard Dependent
  - onKeyDown
  - onKeyUp
- Device independent:
  - onFocus
  - onBlur
  - onSelect
  - onChange
  - **onClick** (ENTER will only work with link or form elements, will not work with non-link and non-control elements, such as plain text, table cells, or <div> elements)



# Key Event KeyCode cheat sheet

[developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/keyCode](https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/keyCode)

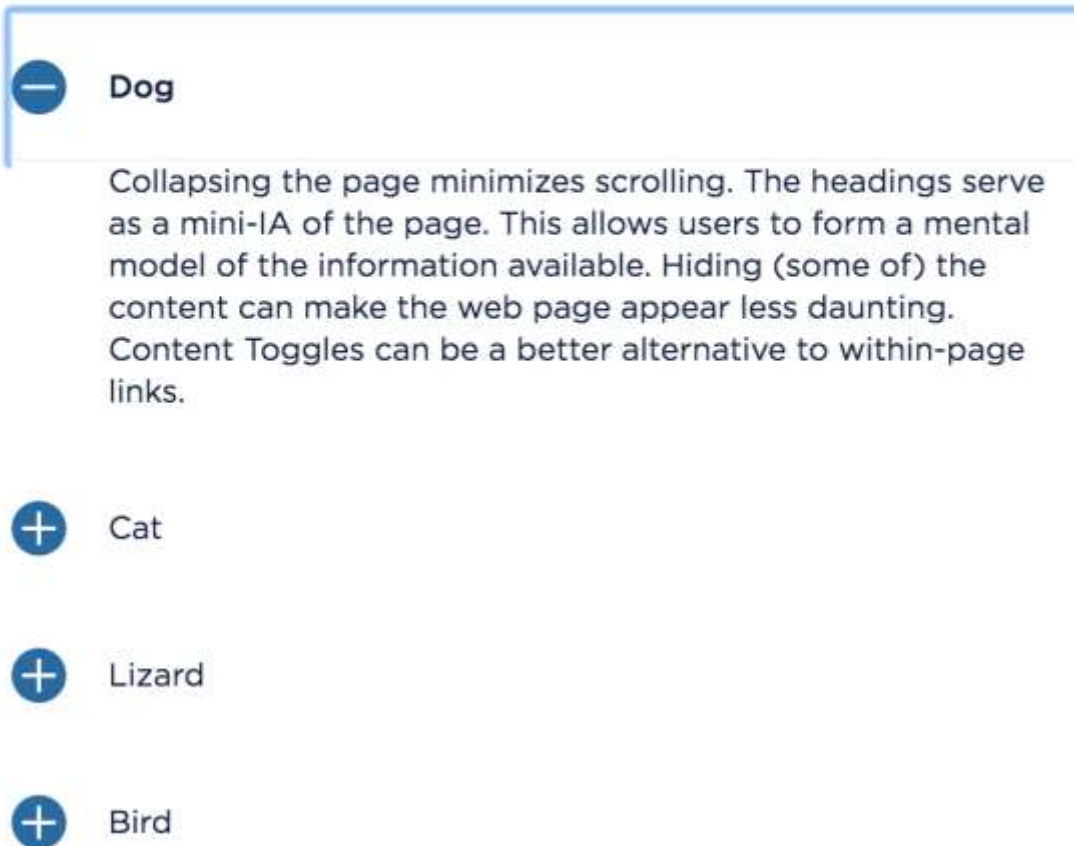
Key	Code
tab	9
enter	13
shift	16
ctrl	17
escape	27
(space)	32
home	36
left arrow	37

(Continued )	
up arrow	38
right arrow	39
down arrow	40
page up	33
page down	34

# Example 1 – Expand/Collapse (accordion)

## Expand Collapse Example:

**Behavior:** Expand/Collapse



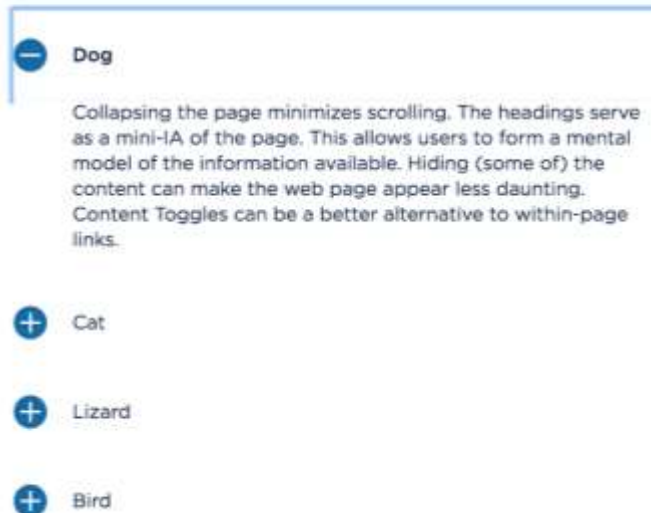
# Example 1 – Expand/Collapse (accordion)

## Expected Screenreader Behavior

- Screen reader reads the accordion toggles state
- When a panel is open state changes and is read to screen reader
- Closed state is hidden from screen reader

### Expand Collapse Example:

Behavior: Expand/Collapse



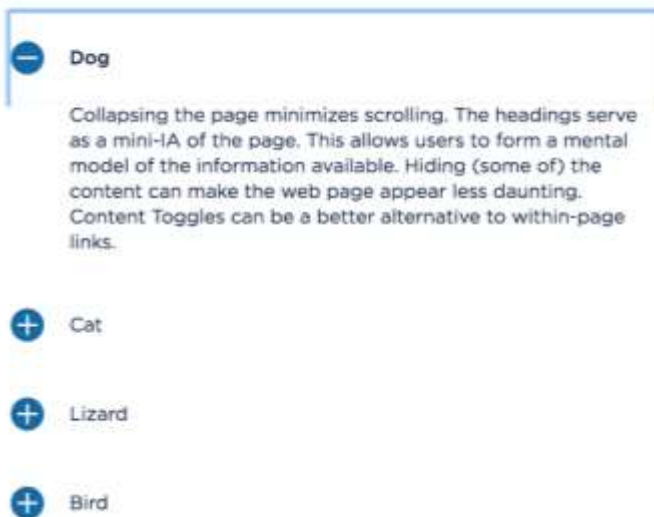
# Example 1 – Expand/Collapse (accordion)

## Expected Keyboard Behavior

- Enter or Space open closes the focused accordion panel
- Down/Up Arrow moves focus to the next or previous accordion panel.
- Home moves focus to the first accordion header.
- End moves focus to the last accordion header.

### Expand Collapse Example:

Behavior: Expand/Collapse

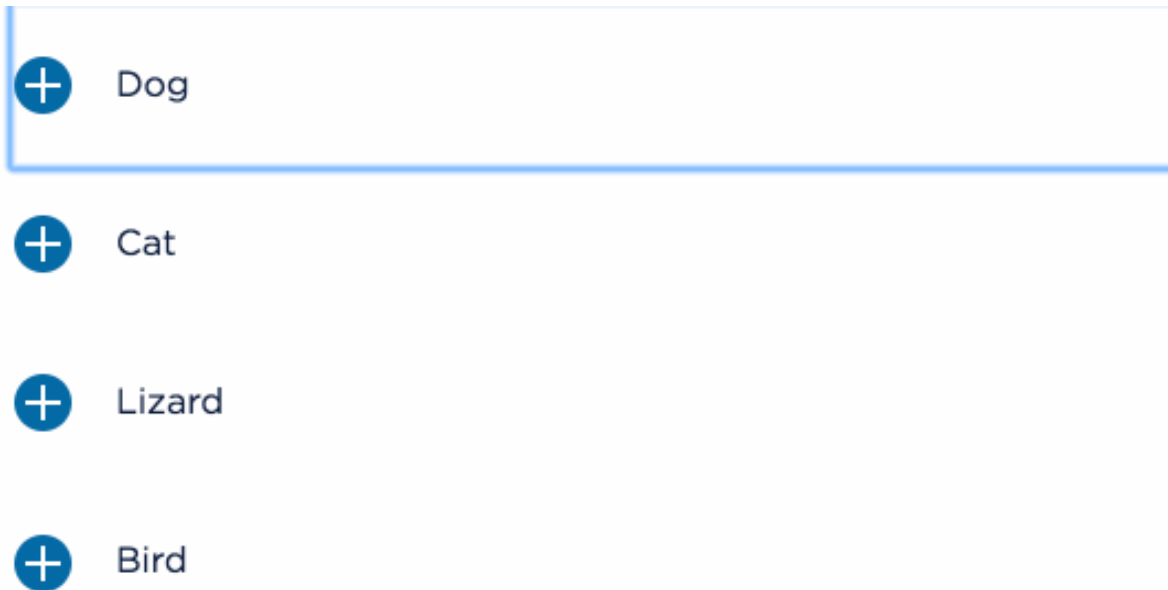


# Keyboard controls to move from accordion panels(sample)

```
function keyboardAccessibility(e) {  
  ...  
  else if (which === 13 || which === 32) { //space and enter  
    OpenAccordion();  
  }  
  else if (which === 38) { //down arrow key  
    if (!accordionActive(target)) {  
      if (FirstAccordionitem(target)) {  
        var toggle = CloseAccordion();  
        toggle.focus();  
      } else {  
        var prevAccordionPanel =  
nextPanel(target, "prev");  
        if (AccordionPanel) {  
          prevAccordionPanel.focus();  
        }  
      }  
    }  
  }  
  ...  
}
```



# Keyboard controls to move from accordion panels(sample)



## Another Accordion

+ Why use a Content Toggle?

## Example 2 - Carousel:



### Title 2

Quisque interdum dictum leo facilisis malesuada. Sed sit amet tortor magna. Vivamus gravida maximus sapien, at convallis nulla tempus quis. Vivamus hendrerit felis at hendrerit tempus.



# Example 2 - Carousel:



## Expected Screenreader Behavior

- Screen reader must announce as a carousel
- Changes to carousel state must be communicated to screen reader users
- All Controls must be read and clear
- Allowing the screen reader user to utilize the keyboard to navigate text within the carousel.
- Pausing the slides so that the user has control of the speed of viewing and minimize distraction.

# Example 2 - Carousel:



## Expected Keyboard Behavior

- Left/Right Arrow to navigate with the dots
- Press TAB and SHIFT+TAB to navigate between arrowed buttons, and press ENTER to activate

# Using Arrow keys to change focus of the navigation (sample)

```
if (key == 13 || key == 32) {
  $carouselSlide.trigger(this, 'click');
}

if (key === 38) { //left key
  if (!dotIndex)
    dotIndex = dotList.length - 1;
  else
    dotIndex--;
} else if (key === 40) { //right key
  if (dotIndex >= dotList.length - 1)
    dotIndex = 0;
  else
    dotIndex++;
}
$carouselObj.query(dotList, function(j, elem) {
  $ carouselObj.setAttr(elem, 'tabindex', '-1');
});
$carouselObj.setAttr(dotList[dotIndex],
'tabindex', '0').focus();
}

...
```

```
<li role="button" tabindex="0" title="Title 2" aria-label="Title 2 of
5 slides" data-index="2" aria-current="true" class="active-
slide"><span aria-hidden="true">2</span></li>
```

## Using Arrow keys to change focus of the navigation (sample)

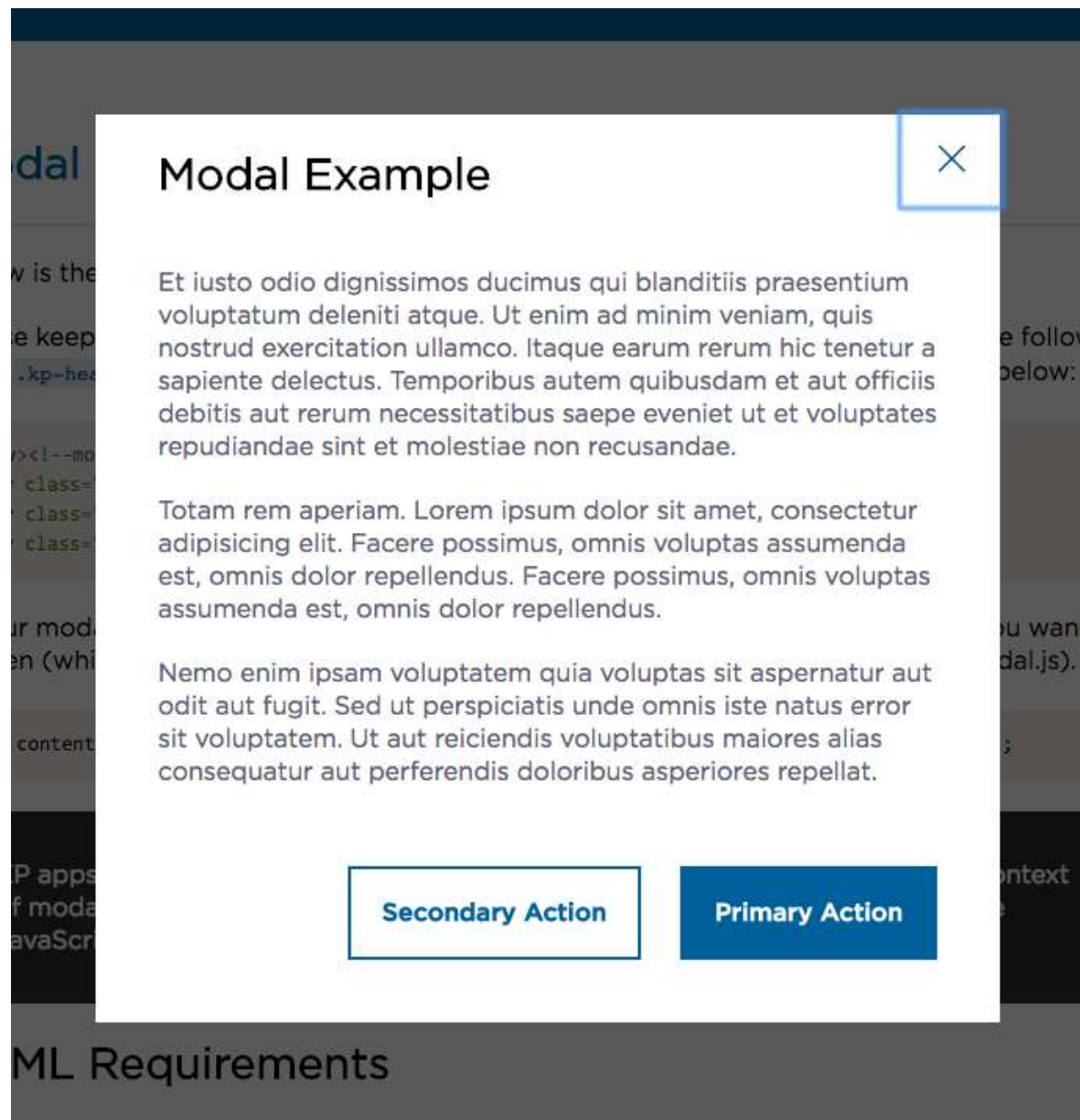
600×300

### Title 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam id ex vel lacus euismod tempus. Ut vulputate pellentesque magna, vitae bibendum enim blandit in.

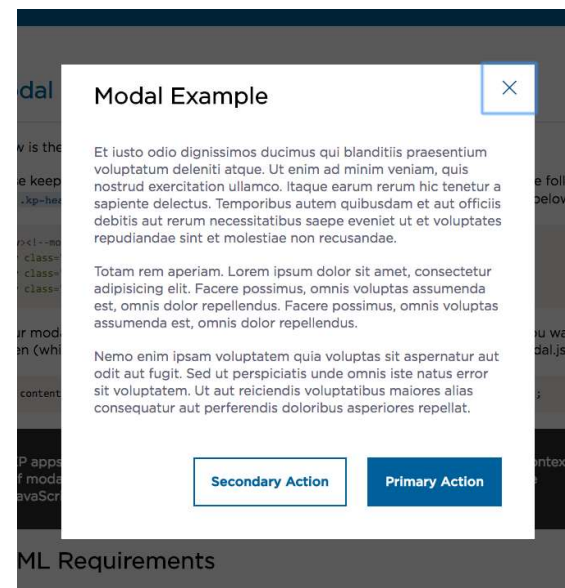


# Example 3 - Modal:





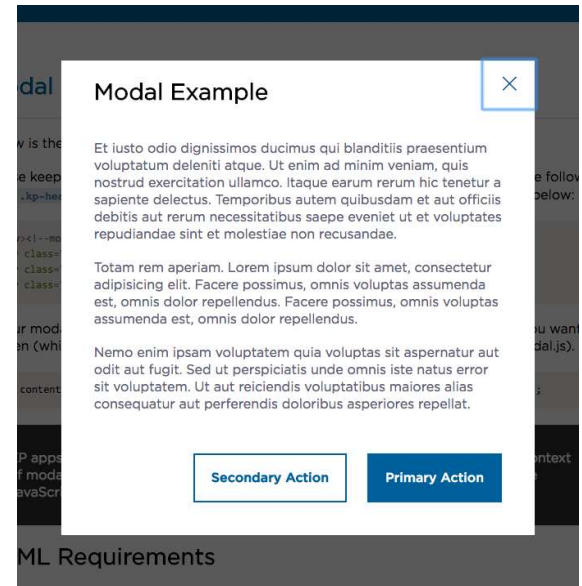
# Example 3 - Modal:



## Expected Screenreader Behavior

- Screen reader reads modal trigger
- When modal opens, contents and role are announced by screen reader
- Does not read, or select, anything outside of the contents of the dialog

# Example 3 - Modal:



## Expected Keyboard Behavior

- SPACE or ENTER opens modal
- TAB or SHIFT+TAB to goes through keyboard trapping
- ESC closes the modal
- When modal is opened, keyboard focus goes to first focused element
- When modal closes focus goes to modal trigger

# Keyboard trapping by using an array of focusable items (sample)

```
$modalOpen.on('keydown', modalKeyboardTrap);  
var $focusableElementsString = $modal.find('button, a, input');  
var $focusableElements = $('modalOpen').find($focusableElementsString);  
var $firstTabStop = $focusableElements[0];  
var $lastTabStop = $focusableElements[$focusableElements.length - 1];  
  
function modalKeyboardTrap(e) {  
  if (e.keyCode === 9) { //Tab Key  
    if (e.shiftKey) {  
      if (document.activeElement === $firstTabStop) {  
        $lastTabStop.focus();  
      }  
    }  
    else {  
      if (document.activeElement === $lastTabStop) {  
        $firstTabStop.focus();  
      }  
    }  
  }  
  if (e.keyCode === 27) { // escape Key  
    closeFullScreenModal();  
  }  
}
```

# Keyboard trapping by using an array of focusable items (sample)

[KP Patterns Library](#) > **Modal Pattern**

## Modal Pattern

**Demo**

[Installation](#)

[Accessibility](#)

[Markup](#)

[JS/jQuery](#)

[Style](#)

### Demo

The Modal pattern, as example below has already been integrated into AEM as a global shared component, there is any additional modifications that needs to be done please contact the styleguide/patterns team

Bring in the modal

This is some content that's behind the modal/carousel. Isn't that pretty nifty?

### Full Screen Takeover

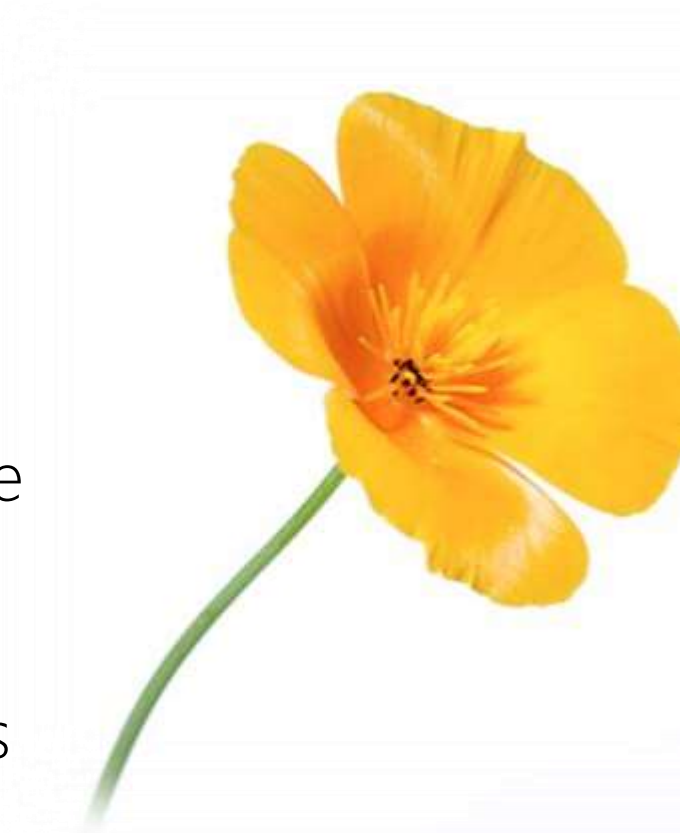
# Benefits for developers

- Meets all Requirements
- Saves development time
- Beginner accessibility developers can leverage compliant code examples
- Extendable code packages saves time, and defects
- Positive user experience



# Summary

- Style Guide and Pattern Library can:
  - saves development time coding interactive web elements from scratch
  - improve accessibility and decrease the defects found during the product life cycle
  - Allows for a UX standard as well as accessibility standard.
  - Meet all requirements



# Break for Questions





# thank you

[www.linkedin.com/in/drmarkstimson](https://www.linkedin.com/in/drmarkstimson)  
[www.linkedin.com/in/vita-zavoli-7a3b0a4/](https://www.linkedin.com/in/vita-zavoli-7a3b0a4/)  
[www.linkedin.com/in/nancycrandall/](https://www.linkedin.com/in/nancycrandall/)





# Appendix

# Additional Example 1 - Color:

Color contrast passes Accessibility requirements

Give clear guidelines to use colors for specific elements

## Main Section Colors

Medical Record

Message Center

Coverage and costs

Appointments

Health and wellness

Pharmacy

Midnight #003371	Forest #6A7CDB	Burnt Sienna #D1480C	Ocean #00669D	Acid #E06654	Plum #472F9F
XP Blue #0066A6	Green #57A65B	Tiger Lily #F9C00F	Aqua #00A79B	Floral #F06674	Libc #606F81
Sky #90CEFF	Lime #AADC33	Tangerine #FF990F	Boon Egg #00D09F	Rose #FFA4B4	Lavender #D7ACDA

Note: Mid/Light values are to be used sparingly and as decorative elements only. Avoid using text on these colors due to accessibility.

## Interactive Color



## Text Colors



Note: Dolphin only meets WCAG AA compliance when on a white (#FFFFFF) or pebble (#F7F7F7) background

## Decorative Element Colors



Note: These colors are generally used for backgrounds, dividers and form elements.

If using as a background with text please adhere to accessibility rules (4.5) color contrast

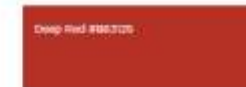
Secondary icon circle color:  
Secondary icons use this new 'Pale Blue' as their background color.

## Disabled Color



Note: Disabled states DON'T need to adhere to color contrast accessibility rules

## Error



# Color Standard Made Simple

```
.link_element:hover {  
background-color:$interactive-hover-color;  
}
```

## Text Colors

Dolphin text color only meets WCAG AA compliance when on a white (#FFFFFF) or pebble (#F7F7F7) background. White text color must be used on dark colors.



inky-blue  
\$text-dark  
hex: #0D1C3D



dolphin  
\$text-light  
hex: #677083

white  
\$white  
hex: #FFFFFF

## Interactive Colors



kp-blue  
\$link-color  
hex: #006BA6



interactive-blue-dark  
\$link-hover-color  
\$button-hover-color  
hex: #003B71