
Auditing Designs Systems for

ACCESSIBILITY

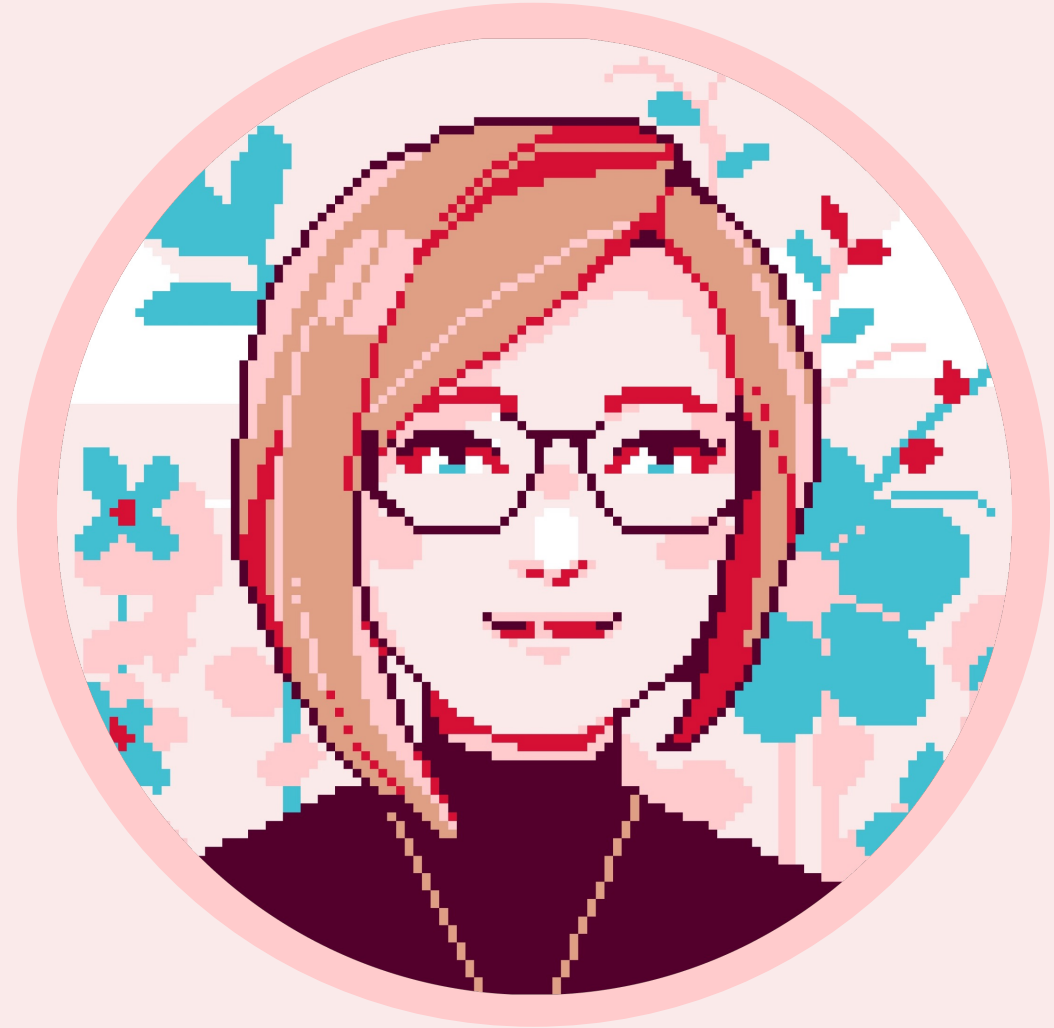
Anna E. Cook

11 / 2022

Hi, I'm Anna!

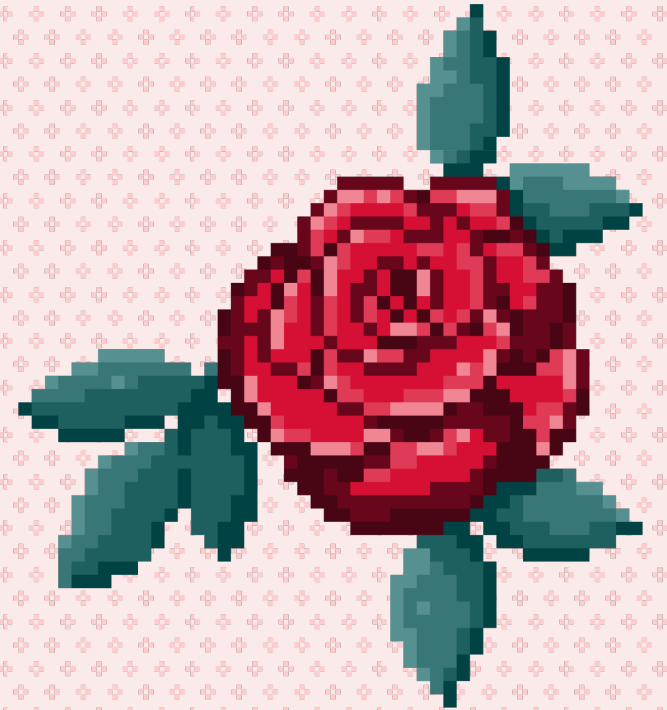
Pronouns are She/They

- Senior Designer at Microsoft working on accessibility and inclusive design for Azure
- Master's student studying inclusive design at the ATLAS Institute of CU-Boulder



What we will discuss

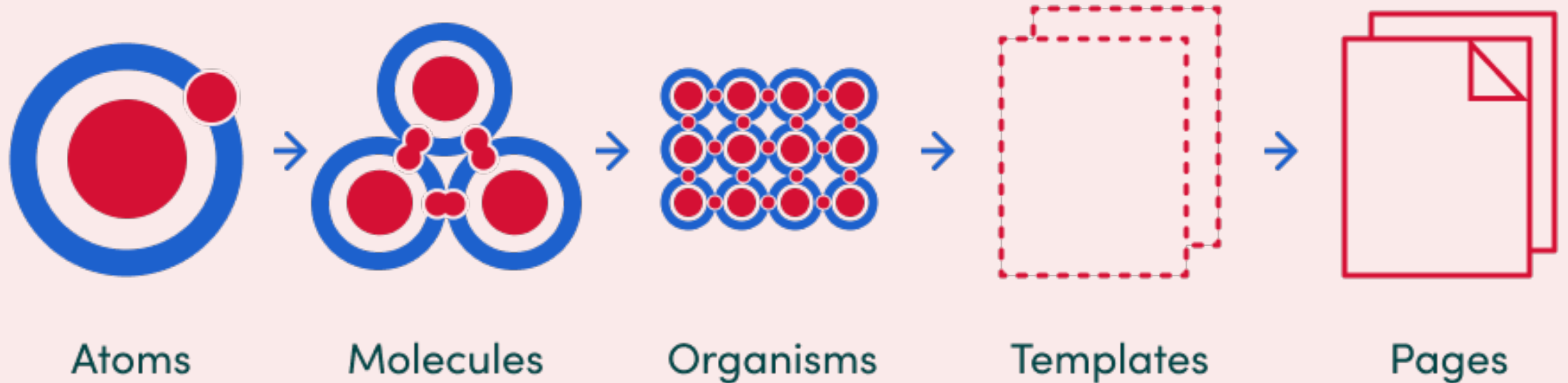
1. Design systems and accessibility
2. Auditing your design system
3. Documenting your audit findings
4. Acting on your accessibility audit



01

Design Systems & Accessibility

Atomic Design Principles



Atoms

Label

Dropdown

BUTTON

Molecules

State ZIP

Georgia 30301

SAVE

Organisms

First Name Last Name

Stacey Abrams

Company

Address Apt, suite

1234 Vote St. #303

City

Atlanta

State Postal Code

GA 30301

Phone Number

Templates & Pages

Shipping Address

First Name Last Name

Stacey Abrams

Company

Address Apt, suite

1234 Vote St. #303

City

Atlanta

State Postal Code

GA 30301

Phone Number

SAVE

Let's look at this system a little more closely...

The screenshot shows a mobile application interface for editing a shipping address. The title bar is blue with a white hamburger menu icon on the left and three small white icons (square, circle, triangle) on the right. The title 'Shipping Address' is in white text. The form is white with a light gray border. It contains several input fields: 'First Name' (Stacey), 'Last Name' (Abrams), 'Company' (empty), 'Address' (1234 Vote St.), 'Apt, suite' (#303), 'City' (Atlanta), 'State' (GA with a dropdown arrow), 'Postal Code' (30301), and 'Phone Number' (empty). A blue 'SAVE' button is at the bottom left.

Field Label	Value
First Name	Stacey
Last Name	Abrams
Company	
Address	1234 Vote St.
Apt, suite	#303
City	Atlanta
State	GA
Postal Code	30301
Phone Number	

Checking the accessibility of our page...

The image shows a mobile application interface for a 'Shipping Address' form. The form is displayed on a white background with a blue header bar. The header bar contains a hamburger menu icon (three horizontal lines) on the left and the text 'Shipping Address' in white. Below the header, the form consists of several input fields. The first two fields are labeled 'First Name' and 'Last Name', with the values 'Stacey' and 'Abrams' respectively. Below these is a 'Company' field, which is currently empty. At the bottom, there are two more fields labeled 'Address' and 'Apt, suite', both of which are also empty. Red boxes highlight the hamburger menu icon, the 'First Name' label, and the 'Company' input field. Red arrows point from these elements to text annotations on the right side of the image.

No icon button label provided (visually or programmatically)

None of the form fields have required indicators, so users wouldn't know what is required or optional.

The form fields have low contrast borders, users may struggle to select the field (contrast ratio is 1.48 to 1).

City

Atlanta

State

Postal Code

GA

30301

Phone Number

SAVE

Will users be able to zoom in on this interface up to 200% without loss of content or functionality?

No input instructions are provided, how will users prevent errors if an input is not matching desired format?

The color contrast on the save button is too low, people will struggle to read it (contrast ratio is 2.38 to 1).

Looking at our button atom

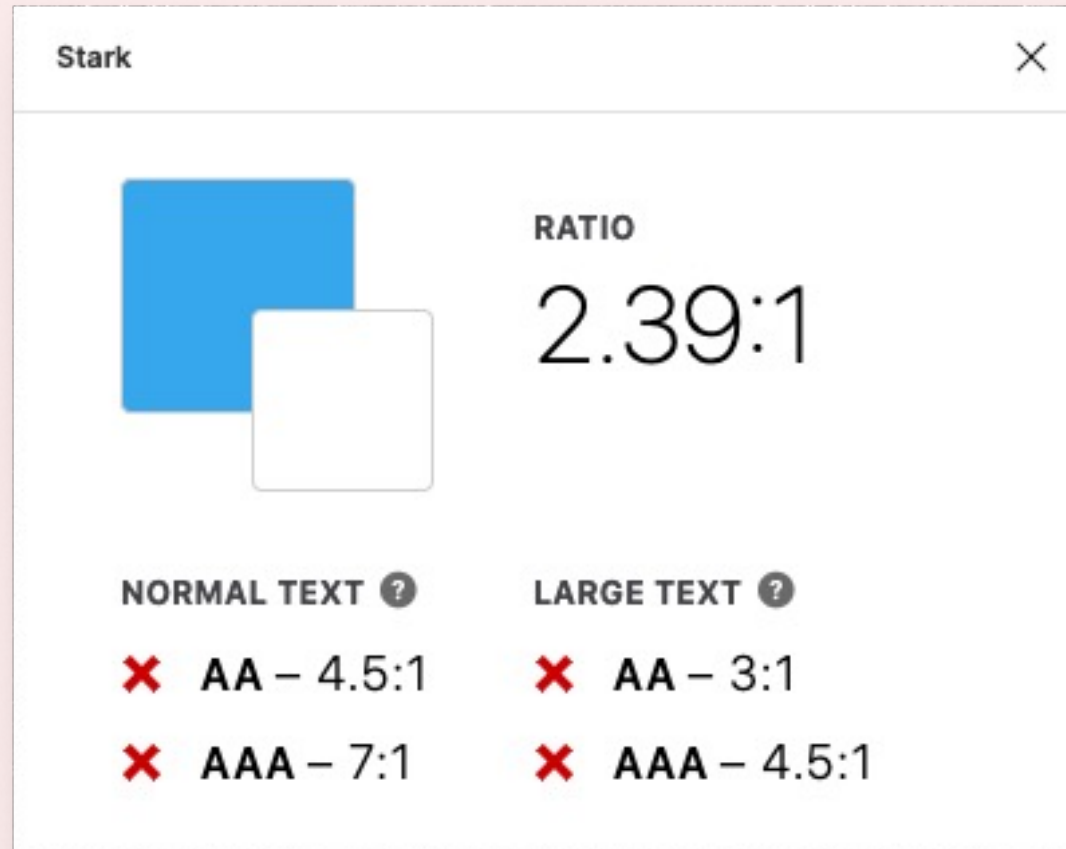


Background color

- #3DA07B

Button Text

- Source Sans Pro
- SemiBold
- 14px
- White



Stark

RATIO

2.39:1


NORMAL TEXT ?

LARGE TEXT ?

✗ AA – 4.5:1	✗ AA – 3:1
✗ AAA – 7:1	✗ AAA – 4.5:1

The image shows a Stark accessibility checker window. It displays a blue square and a white square, representing the button's background and text. The contrast ratio is shown as 2.39:1. Below this, there are two columns of text: 'NORMAL TEXT' and 'LARGE TEXT'. Each column has two rows of contrast ratios, all marked with a red 'X' to indicate they do not meet WCAG standards. The 'NORMAL TEXT' column shows AA – 4.5:1 and AAA – 7:1. The 'LARGE TEXT' column shows AA – 3:1 and AAA – 4.5:1.

This issue affects other components



Heading Inline Action

PRICE

Monthly fee:	\$299.00 USD
Estimated tax:	\$0.00 USD
Order total:	\$299.00 USD

BUTTON

Two line text string with two actions.
One to two lines is preferable.

ACTION ACTION

×

Tag Label

Tag Label

This is a heading that introduces the tooltip

This is the body content that explains what the tooltip is about, and gives you the ability to learn more.

Skip **NEXT**

Creating a more
accessible design system
empowers teams to
build inclusive products

02

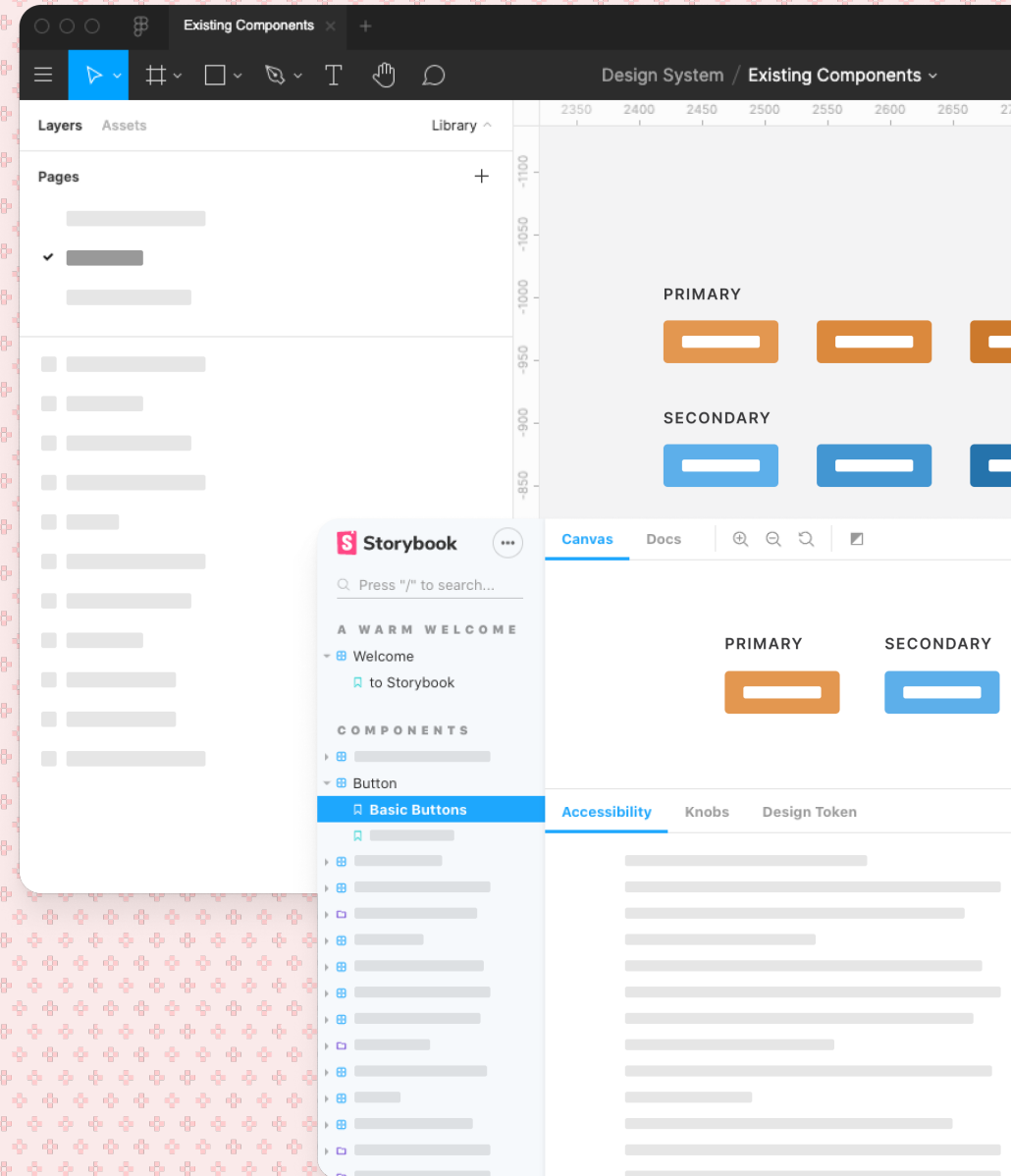
Auditing your design system

Accessibility audits are a way to find and log issues in our designs so we can fix them

STEP 1

Log components in audit for review

Review components most actively used to create products in both designs and code.



Start your audit documentation

Item
Button (Primary)
Cards
Date Picker
Dialog
Lists

Reviewer(s)	Anna E. Cook
Summary	This report describes the conformance of Ant Design's system with W3C's Web Content Accessibility Guidelines (WCAG). Our goal is that the website should substantially conform to Level AA compliance as defined by the Web Content Accessibility Guidelines (WCAG) 2.1, level AA.
Scope of Review	Button - Ant Design
URLs	https://ant.design/components/button/
Timeline	3/10/2021
Review Process	<ul style="list-style-type: none">- Conformance evaluation of web accessibility requires a combination of semi-automated evaluation tools and manual evaluation by an experienced reviewer. Automated testing can be used to establish a baseline of accessible functionality, manual testing can be used to understand greater accessibility opportunities. Evaluation and validation tools used:<ul style="list-style-type: none">- Deque's axe plugin (https://www.deque.com/axe/) was used within chrome to run automated testing as well as guided manual testing- Manual testing was performed using the WCAG Quick Reference Guide (https://www.w3.org/WAI/WCAG21/quickref/) for items that were not considered in automated testing.

STEP 2

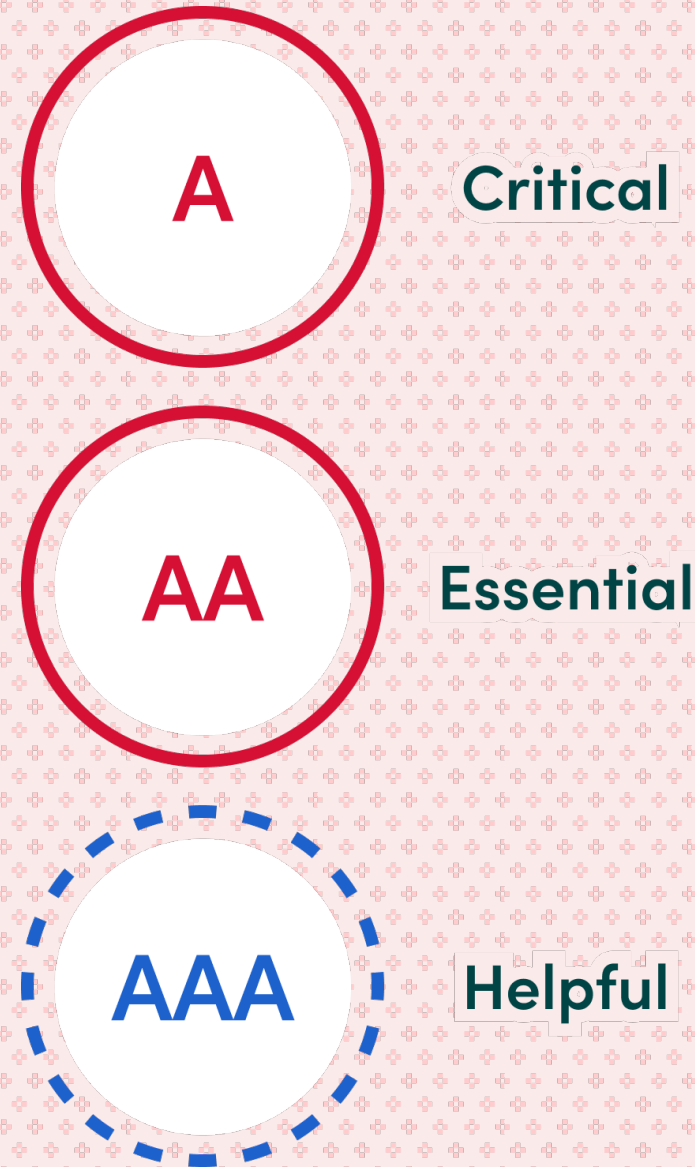
Use WCAG to check designs and code

The Web Content Accessibility

Guidelines are globally validated

standards for digital accessibility and

essential to your audit.



WCAG's Quick Reference Guide can serve as a checklist for your review

2. Operable **Guiding principle**

- 2.1 Keyboard Accessible
 - 2.1.1 Keyboard
 - 2.1.2 No Keyboard Trap
 - 2.1.3 Keyboard (No Exception)
 - 2.1.4 Character Key Shortcuts
- 2.2 Enough Time
 - 2.2.1 Timing Adjustable
 - 2.2.2 Pause, Stop, Hide
 - 2.2.3 No Timing
 - 2.2.4 Interruptions
 - 2.2.5 Re-authenticating
 - 2.2.6 Timeouts
- 2.3 Seizures and Physical Reactions
 - 2.3.1 Three Flashes or Below Threshold
 - 2.3.2 Three Flashes
 - 2.3.3 Animation from Interactions
- 2.4 Navigable
 - 2.4.1 Bypass Blocks
 - 2.4.2 Page Titled
 - 2.4.3 Focus Order
 - 2.4.4 Link Purpose (In Context)
 - 2.4.5 Multiple Ways
 - 2.4.6 Headings and Labels
 - 2.4.7 Focus Visible
 - 2.4.8 Location
 - 2.4.9 Link Purpose (Link Only)
 - 2.4.10 Section Headings
- 2.5 Input Modalities

Guideline 2.1 – Keyboard Accessible

Make all functionality available from a keyboard.

Guideline **Acceptance level**

2.1.1 Keyboard — **Level A**

All functionality of the content is operable through a keyboard interface without requiring specific timings for individual keystrokes, except where the underlying function requires input that depends on the path of the user's movement and not just the endpoints.

Note 1: This exception relates to the underlying function, not the input technique. For example, if using handwriting to enter text, the input technique (handwriting) requires path-dependent input but the underlying function (text input) does not.

Note 2: This does not forbid and should not discourage providing mouse input or other input methods in addition to keyboard operation.

[Understanding 2.1.1](#)

[Show techniques and failures for 2.1.1](#)

[← SHARE](#) | [↑ BACK TO TOP](#)

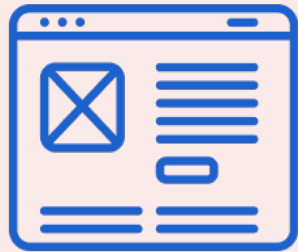
2.1.2 No Keyboard Trap — Level A

If keyboard focus can be moved to a component of the page using a keyboard interface, then focus can be moved away from that component using only a keyboard interface, and, if it requires more than unmodified arrow or tab keys or other standard exit methods, the user is advised of the method for moving focus away.

Note 1: Since any content that does not meet this success criterion can interfere with a user's ability to use the whole page

What to review in your audit

Designs



Code



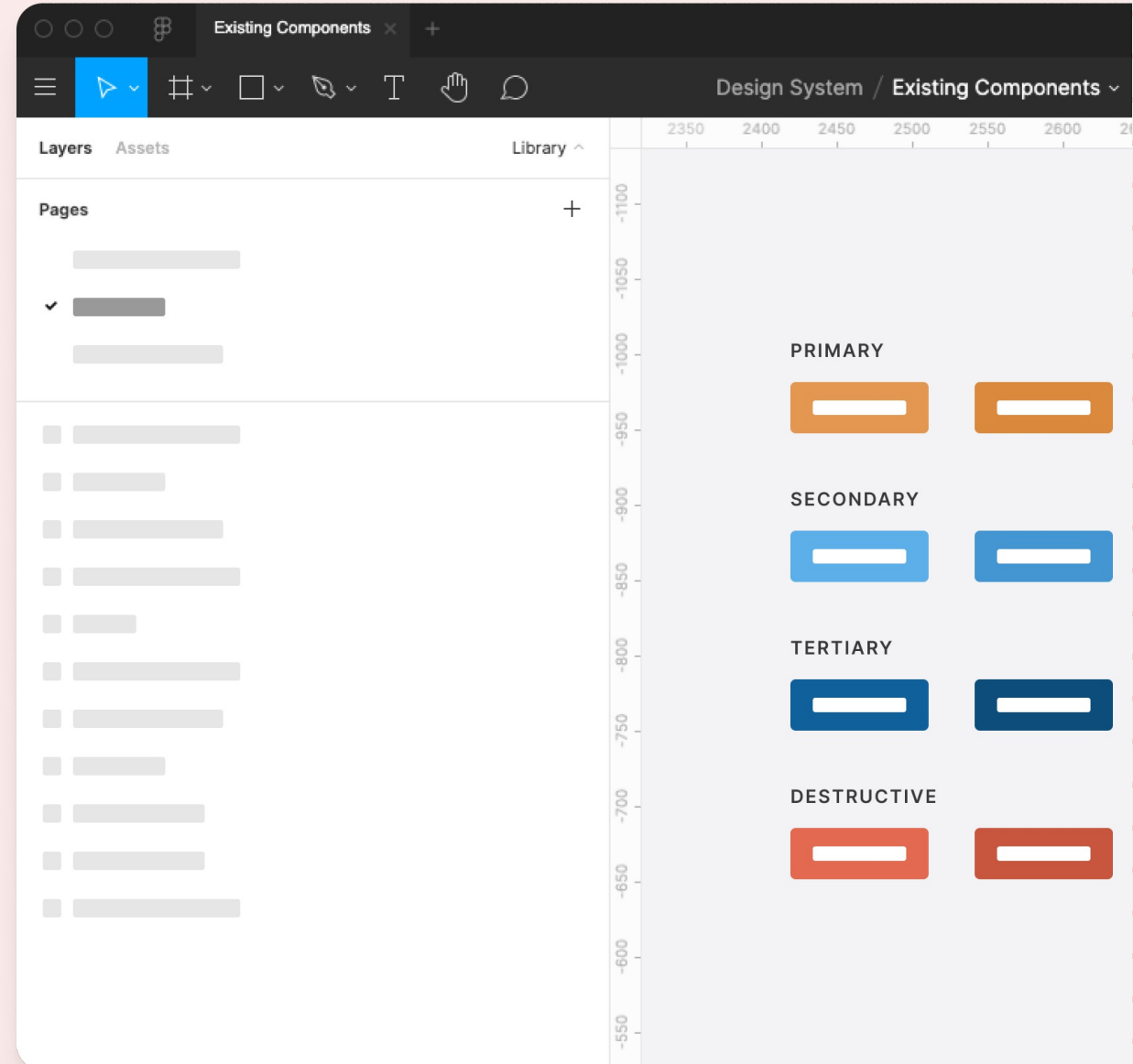
Docs



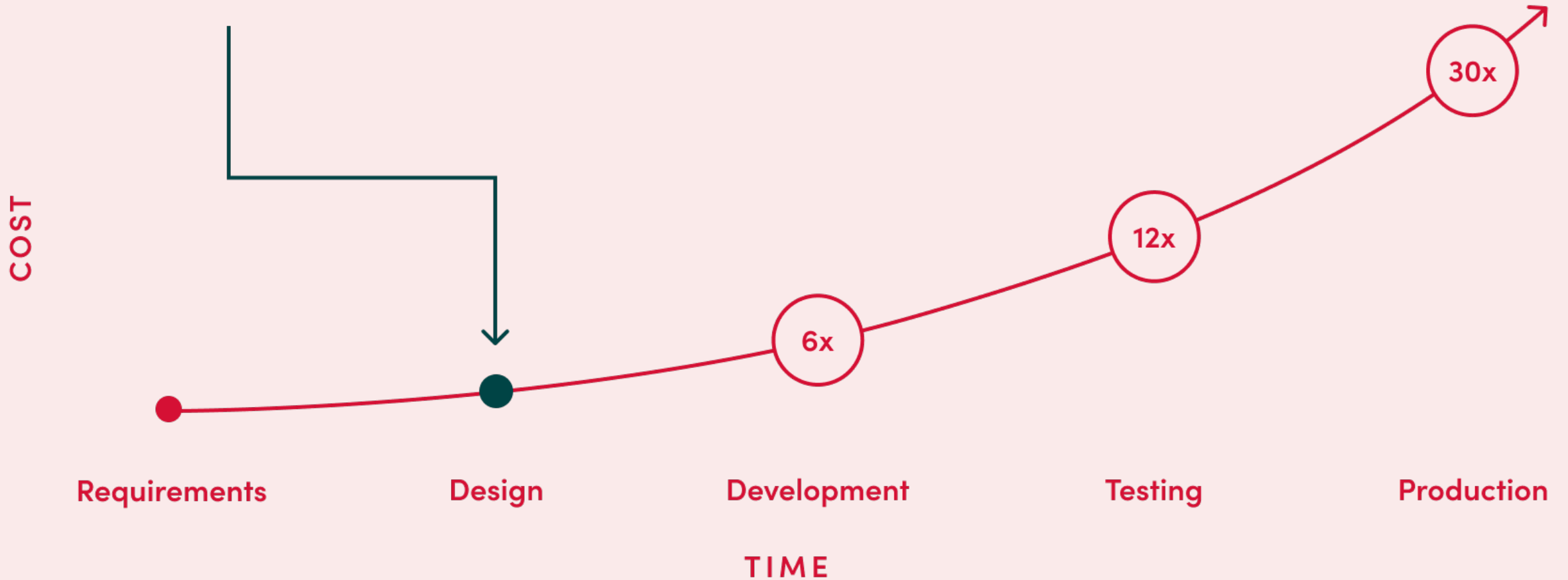
STEP 3

Review designs

Developers can't out code inaccessible designs and many issues start there.



67% of accessibility issues can originate in design*



* Percentages of design issues may vary, this stat is not an exact representation of all audits

What do we review in design?

- Color contrast and usage
- Content, copy, readability
- Headings and Page Titles
- Link purpose
- Hover and focus states
- Forms (errors, labels, etc.)
- Layout (consistency, responsiveness)
- Media (captions, alt text, etc.)
- Tab order and bypass blocks
- Timing
- Typography

Look past styling...

Relies on color alone
(WCAG 1.4.1 Use of Color)


This provides information

This is a success message

This is a warning


This shows an error

Communicates more meaning
with icon and alt text

 This provides information

 This is a success message

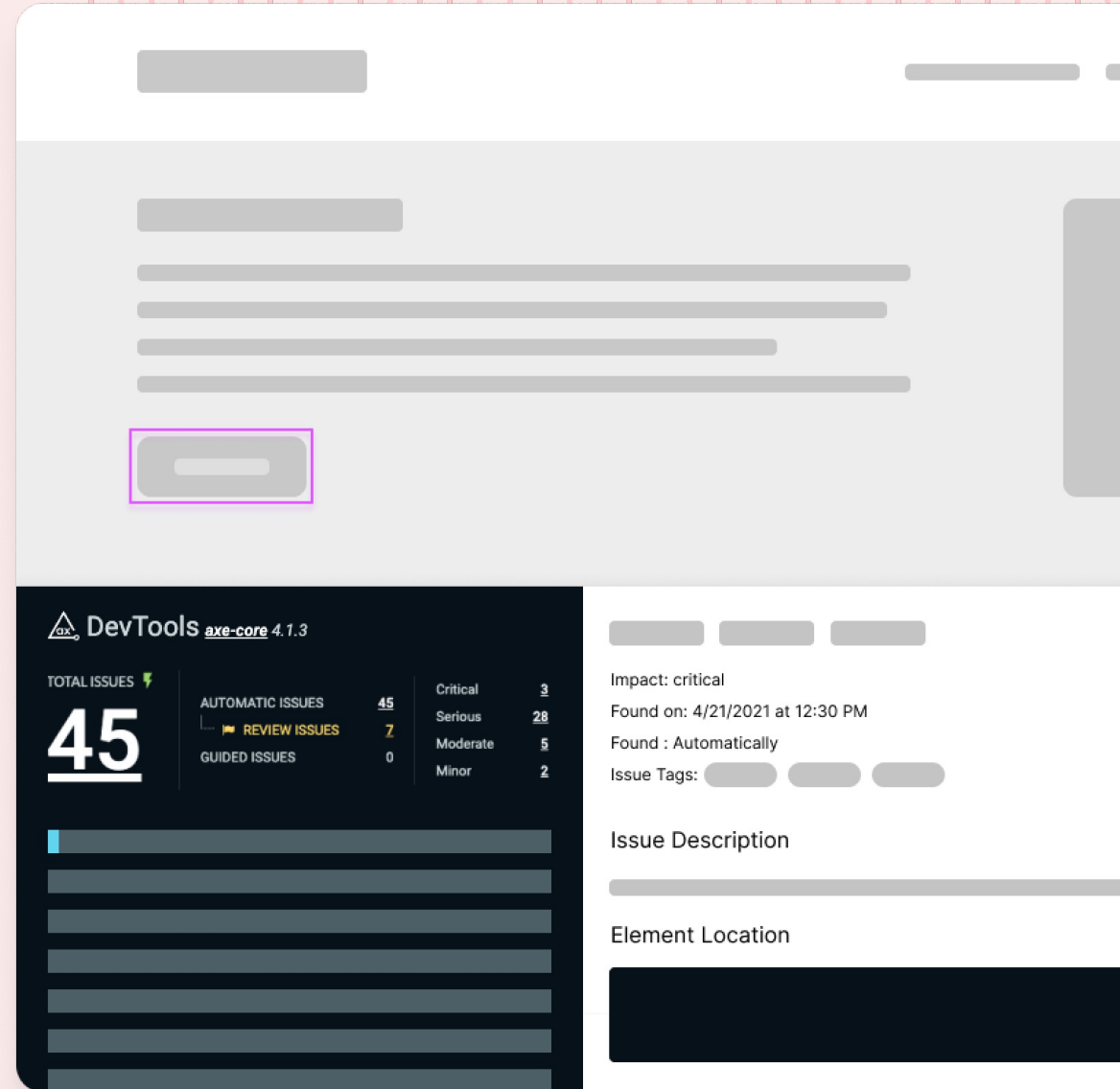
 This is a warning

 This shows an error

STEP 4

Review Code

Developers should use tools built into their browser and linter extensions to review code



STEP 5

Review System Documentation

Even with a perfectly accessible component library, consuming teams can break product accessibility by using components incorrectly.

Accordion

Usage

Style

Code

Accessibility

Design annotations are needed for specific instances shown below, but for the standard accordion component, Carbon already incorporates accessibility.

- ↳ What Carbon provides
- ↳ Design recommendations
- ↳ Development considerations

What Carbon provides

Carbon bakes keyboard operation into its components, improving the experience of blind users and others who operate via the keyboard. Carbon incorporates many other accessibility considerations, some of which are described below.

STEP 6

Add the issues you find to your doc

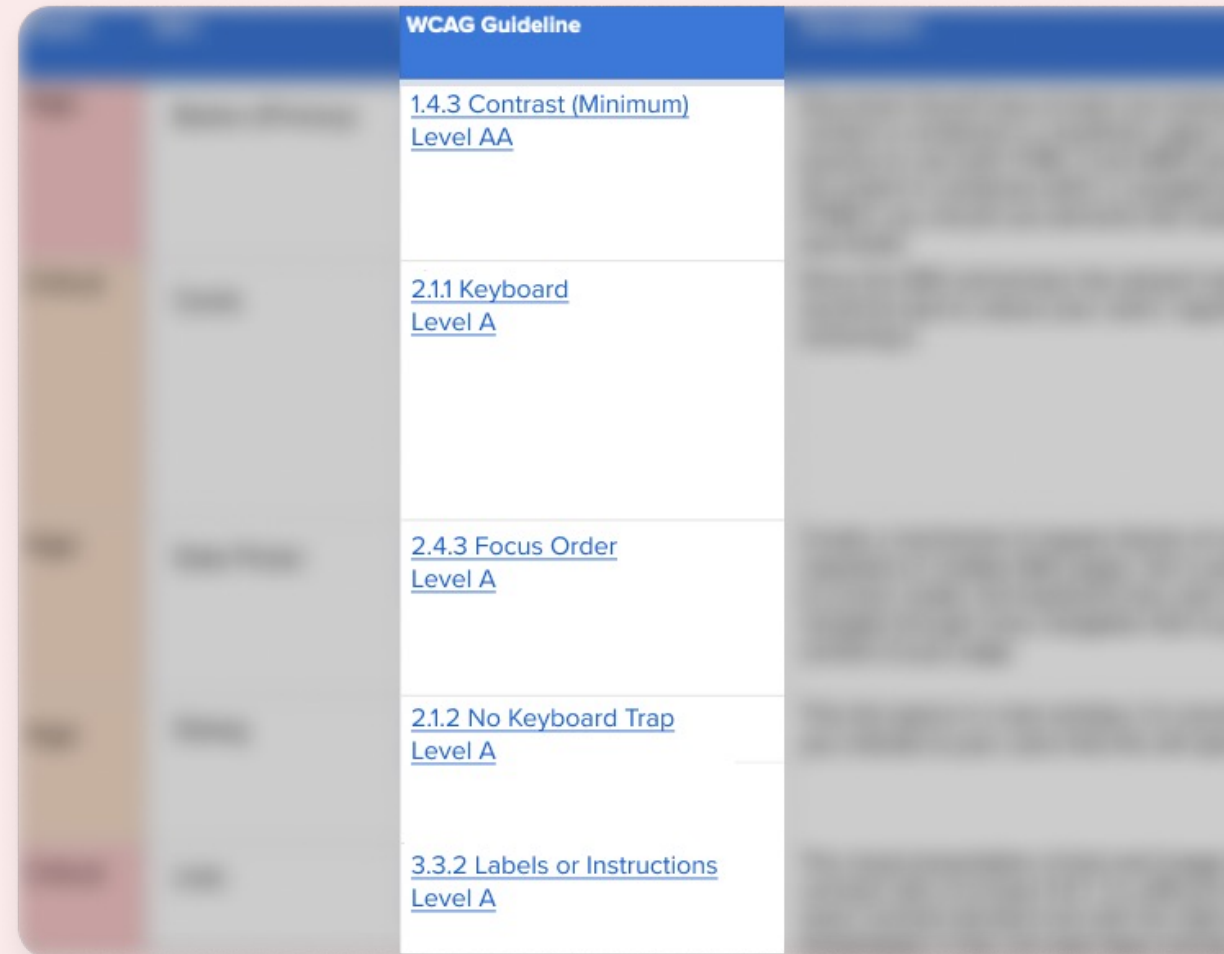
Audit docs should be actionable and easy to reference back to later.

Impact	Item	WCAG Guideline
High	Button (Primary)	1.4.3 Contrast (Minimum) Level AA
Critical	Cards	2.1.1 Keyboard Level A
High	Date Picker	2.4.3 Focus Order Level A
High	Dialog	2.1.2 No Keyboard Trap Level A
Critical	Lists	3.3.2 Labels or Instructions Level A

Map issues to WCAG criteria

Using WCAG to do your audit will make it easy to link what you've found to specific guidelines. This will also help...


- Prioritize issues (A > A > AAA)
- Provide references for whom ever fixes the issue later
- Clarify to leadership that issues found are related to legitimate guidelines



The image shows a screenshot of a table with a blue header row labeled "WCAG Guideline". The table contains five rows, each with a colored background (red, orange, yellow, green, and purple) and a corresponding WCAG guideline reference. The references are: 1.4.3 Contrast (Minimum) Level AA, 2.1.1 Keyboard Level A, 2.4.3 Focus Order Level A, 2.1.2 No Keyboard Trap Level A, and 3.3.2 Labels or Instructions Level A.

WCAG Guideline	
	1.4.3 Contrast (Minimum) Level AA
	2.1.1 Keyboard Level A
	2.4.3 Focus Order Level A
	2.1.2 No Keyboard Trap Level A
	3.3.2 Labels or Instructions Level A

Group issues into common themes



Heading [Inline Action](#)

PRICE

Monthly fee:	\$299.00 USD
Estimated tax:	\$0.00 USD
Order total:	\$299.00 USD

Button

Two line text string with two actions.
One to two lines is preferable.

[ACTION](#) [ACTION](#)

×

Tag Label

Tag Label

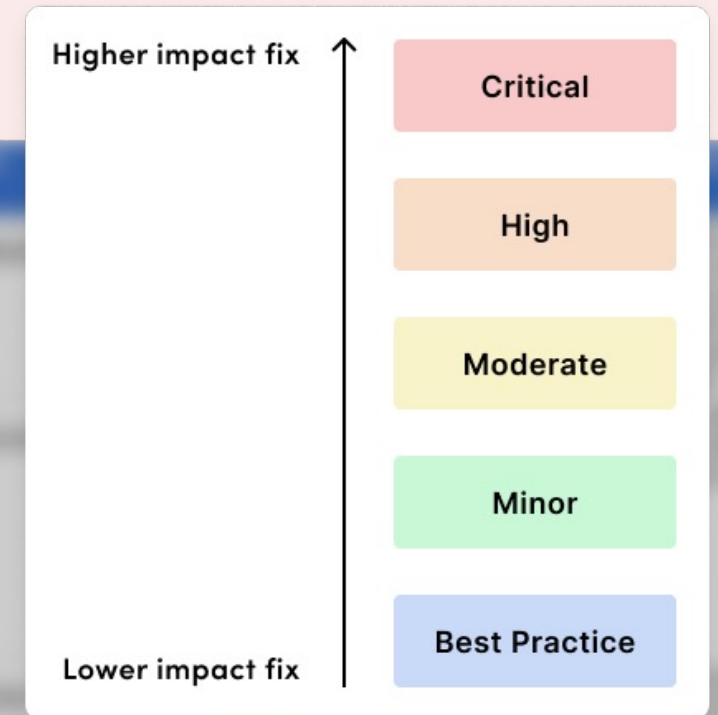
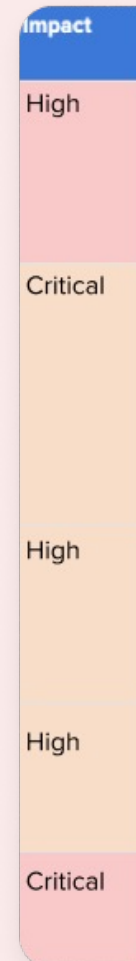
This is a heading that introduces the tooltip

This is the body content that explains what the tooltip is about, and gives you the ability to learn more.

Skip **NEXT**

Prioritize issues by impact

- **Impact on users** – how significant of a barrier is created with this issue?
- **Impact on business** – does this component get used in essential interactions in our business (example: a log in form to access product)?
- **Impact by usage** – how often is this component used across products?



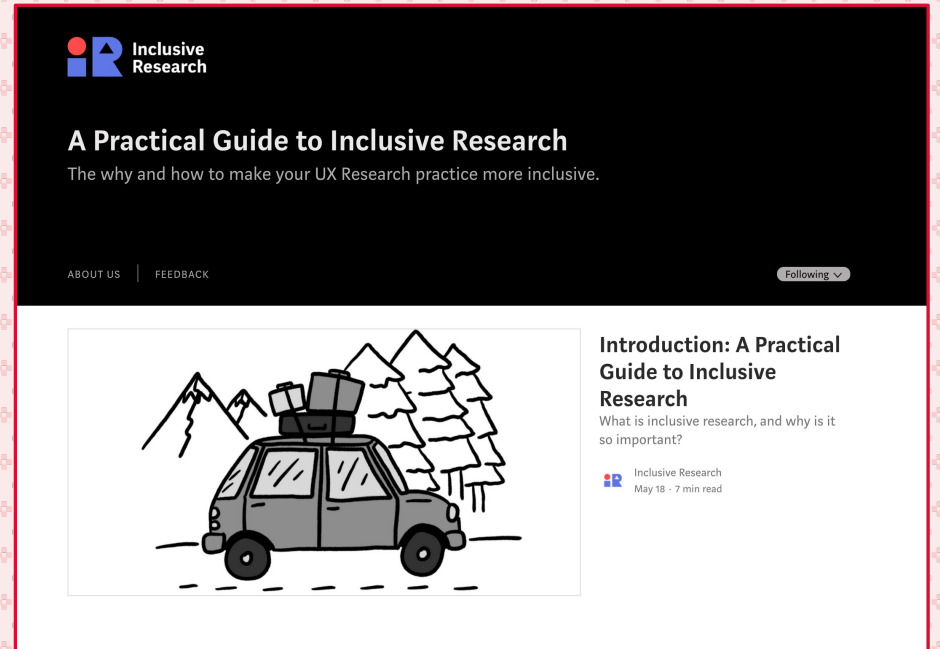
Adjust audit as needed
for the system, product,
and organization

Auditing cannot replace
testing and designing
with disabled users.

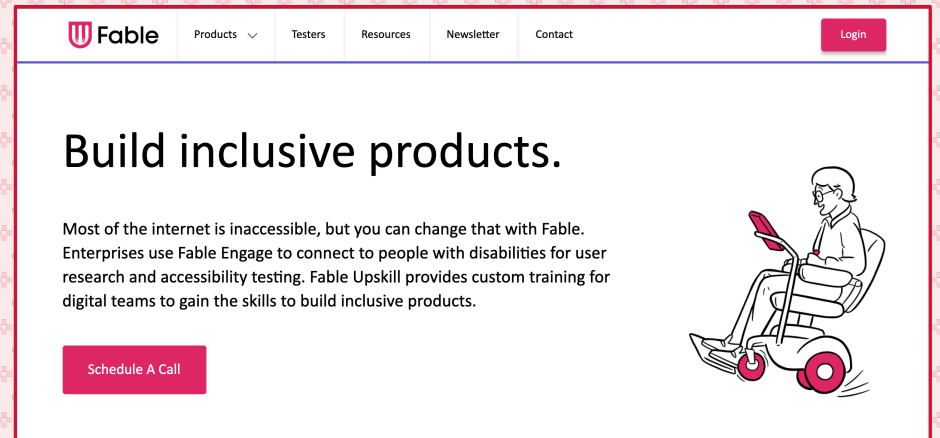
Look into inclusive research strategies

Gather direct feedback from disabled users about your product.

- [Practice inclusive research](#)
- [Conduct inclusive testing](#)



The screenshot shows the top portion of a webpage. At the top left is the Inclusive Research logo, consisting of a stylized 'R' with a blue and red dot. To its right is the text 'Inclusive Research'. Below this is the main heading 'A Practical Guide to Inclusive Research' in a bold, white font. Underneath the heading is a subtitle: 'The why and how to make your UX Research practice more inclusive.' There are two small links, 'ABOUT US' and 'FEEDBACK', and a 'Following' button with a dropdown arrow. The main content area features a large illustration of a grey SUV with a roof rack full of gear, parked on a road with mountains and trees in the background. To the right of the illustration is the article title 'Introduction: A Practical Guide to Inclusive Research' and a short introductory paragraph: 'What is inclusive research, and why is it so important?'. Below the text is the Inclusive Research logo again, followed by the date 'May 18' and '7 min read'.



The screenshot shows the homepage of the Fable website. At the top left is the Fable logo, a stylized 'F' inside a square. To its right are navigation links: 'Products', 'Testers', 'Resources', 'Newsletter', and 'Contact'. On the far right is a red 'Login' button. The main heading is 'Build inclusive products.' Below this is a paragraph of text: 'Most of the internet is inaccessible, but you can change that with Fable. Enterprises use Fable Engage to connect to people with disabilities for user research and accessibility testing. Fable Upskill provides custom training for digital teams to gain the skills to build inclusive products.' At the bottom left is a red 'Schedule A Call' button. On the right side is an illustration of a person with glasses sitting in a motorized wheelchair, using a laptop.

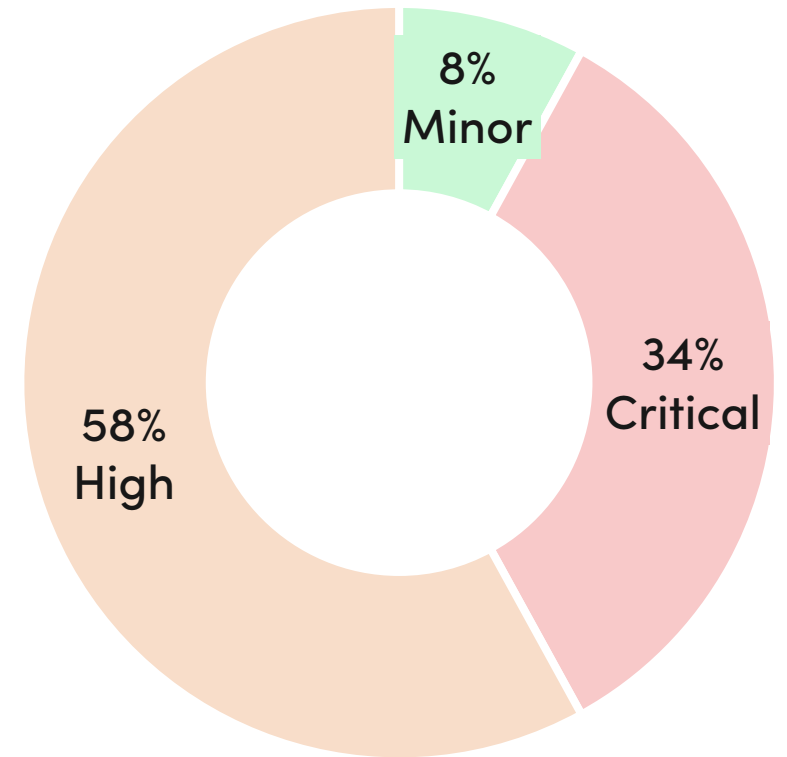
03

Acting on your accessibility audit

Share out results of the
audit with leaders and
consuming teams.

Present a summary of audit findings

A simple summary of your audit will help people understand what you found and what should come next.



Share what issues exist
with consuming teams
using *your system docs*.

Share known issues

Pretending our design system is perfect is a disservice to everyone.

- Let teams know what issues are known in documentation
- Feel free to link to open tickets in your backlog so that teams can
- Create avenues for more feedback
- Create opportunities to contribute and refine

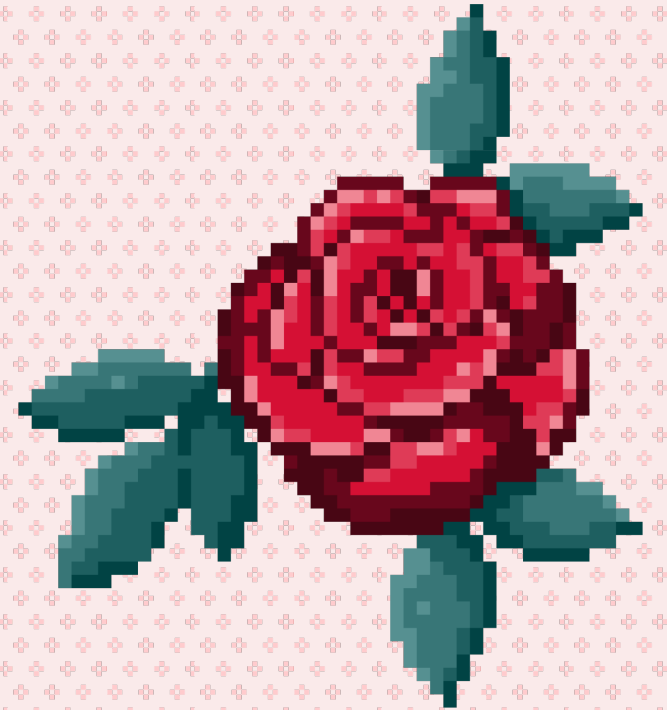
If you leave those
accessibility issues in
your backlog forever,
I will haunt you.



Thank you!

Let's Connect.

www.AnnaECook.com or [@AnnaECook](https://www.instagram.com/AnnaECook)



04
==

Resources & Credits

Tools to get started

- Auditing Design Systems for Accessibility: deque.com/blog/auditing-design-systems-for-accessibility/
- Atomic Design by Brad Frost: atomicdesign.bradfrost.com
- WCAG Quick Reference: w3.org/WAI/WCAG21/quickref
- A11y Project Accessibility checklist: a11yproject.com/checklist
- Stark plugin: getstark.co
- Deque axe: deque.com/axe

Sources

- Slide 7: Atomic Design by Brad Frost atomicdesign.bradfrost.com
- Slide 8: WCAG Quick Reference w3.org/WAI/WCAG21/quickref
- Slide 11: Stark plugin: getstark.co
- Slide 22: Deque axe deque.com/axe
- Slide 21: Deque [Accessibility and Automation: Shift Left ROI](#)
- Slide 21: Deepsources [The exponential cost of fixing bugs](#)
- Slide 23: Carbon Design System carbondesignsystem.com