Treasury Board of Canada Secretariat
Secrétariat du Conseil du Trésor du Canada

Canada

GC Design System

Insights to build an alpha documentation site

October 2021

Index

Executive summary	● Background ● Methodology ○ What, who, how ● Summary of findings
Detailed findings	● Global guidance findings ○ Navigation, search, visual foundations, writing, installation ● Component findings ○ Design, content, code, versioning ● Rough prototype with recommendations ● Annexes

Background

- The GC Design Product Team is working on the **V1 of the GC Design System**
- We wanted to get **good hypothesis to build an alpha documentation site** of the GC Design System
- We asked research participants to try **different tasks** on the Canada.ca design system and other design systems (from both government and private sector)
- We **analyzed what worked and what didn’t** in terms of **navigation**, **search**, **guidance** (content, design and code), and **installation**
- From these insights, we came up with a **set of evidence-informed starting points** to build an alpha documentation site.

Methodology: What

Global guidance

- Navigation (landing page, global, in-page)
- Search
- Visual foundations (typography, colours)
- General content guidance (writing guidelines)
- Installation of the design system

Component guidance

- Design guidance
- Content guidance
- Code guidance
- Versioning and change history

Methodology: Who

- 17 participants (15 EN, 2 FR)
  - 5 designers
  - 5 developers and publishers
  - 7 communicators
- Participants were a mix of *loyal implementers* and *fearless innovators*
- Asked to complete **4 different tasks**, each in **3 different design systems**
- Tasks and design systems were **randomized**

Methodology: How

- We asked people to complete a list of simple, **realistic tasks** that cover a wide range of what people try to do in design systems (tasks were **informed by analytics and previous user interviews**)
- We observed **how they tried to complete the tasks**, noting what they clicked on, where they hesitated, etc.
- We took note of the **time** it took to get them to the **page where the answer was** (if they did)
- We noted **if they found the right answer** (and the time it took to find it)
- We observed and noted what made people succeed (get the right answer) **- that's what worked -** or fail (not getting the right answer) **- that's what didn't work**
- from these observations we **derived a set of starting points** to build an alpha documentation site - picking and choosing what seemed to work best with participants

Methodology: How

Communicators and designers test*

- How to capitalize headings (general content guidance)
- How to write good button text (component content guidance)
- Font and font size (visual foundations)
- Choosing the right alert variant (design guidance)

Developers test**

- Class to use to make radio buttons horizontal (component code guidance)
- Primary colour to use in a new design (visual foundations)
- How to install the design system (installing)
- Identify the last change to the button component (change history)

*Project sheet - Design system testing - Comms and design
**Project sheet - Design system testing - Devs

Findings - Global guidance

- A **persistent left navigation**, with clear and distinct main sections can improve discoverability (especially if with the ability to expand sub-sections and explore)
- A **persistent left search/filter** can improve findability of specific elements
- As a starting point, these main sections could work - more research needed:
  - **Get started** (installation, onboarding, etc.)
  - **Visual foundations** (brand, design tokens, etc.)
  - **Components and patterns**
  - **Writing** (general content guidance)
  - **Community** (how to contribute, contact, etc.)
  - **Latest updates**

Findings - Component guidance

- Component pages need to make it easy to find all these:
    - **design** guidance
    - **content** guidance
    - **code** guidance
- Content guidance need to exist both as **global content guidance** (in the Writing section) and as **component-specific** content guidance

These evidence-informed **starting points** could be used to build the **backbone** of the alpha documentation site of the GC design system.

We could start **adding** tokens and components to that backbone, starting with **foundational visual** elements, **form** elements and important **components**.

Recommendations from findings

We turned the findings into a set of evidence-informed recommendations for building the alpha documentation site:

- Detailed recommendations (slides 43 to 48)
- Figma prototype visualizing the recommendations

Important note: the purpose of the Figma prototype is not to propose a specific design; it's to illustrate some of the recommendations that stem from the findings.

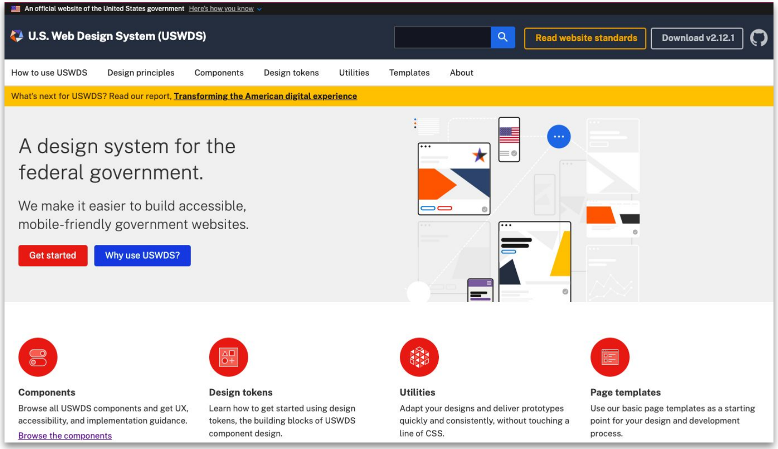
Global guidance findings

Navigation (Landing page, global, in page)
Search
Visual foundations
Writing guidelines
Installation

Landing page - What didn’t work

Tasks: all

✕ Busy landing pages with animation or saturation of elements. Complex navigation menus with many overlapping sections. Video



What's next for USWDS? Read our report, **Transforming the American digital experience**

A design system for the federal government.

We make it easier to build accessible, mobile-friendly government websites.

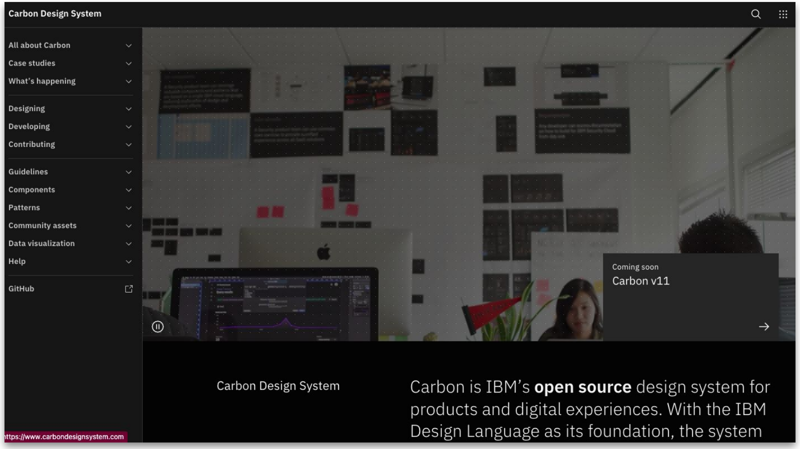
Components
Browse all USWDS components and get UX, accessibility, and implementation guidance.
Browse the components

Design tokens
Learn how to get started using design tokens, the building blocks of USWDS component design.

Utilities
Adapt your designs and deliver prototypes quickly and consistently, without touching a line of CSS.

Page templates
Use our basic page templates as a starting point for your design and development process.

USWDS



Carbon Design System

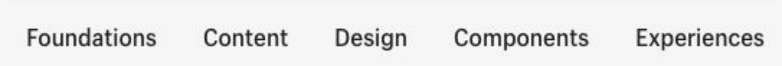
Carbon is IBM’s **open source** design system for products and digital experiences. With the IBM Design Language as its foundation, the system

Carbon

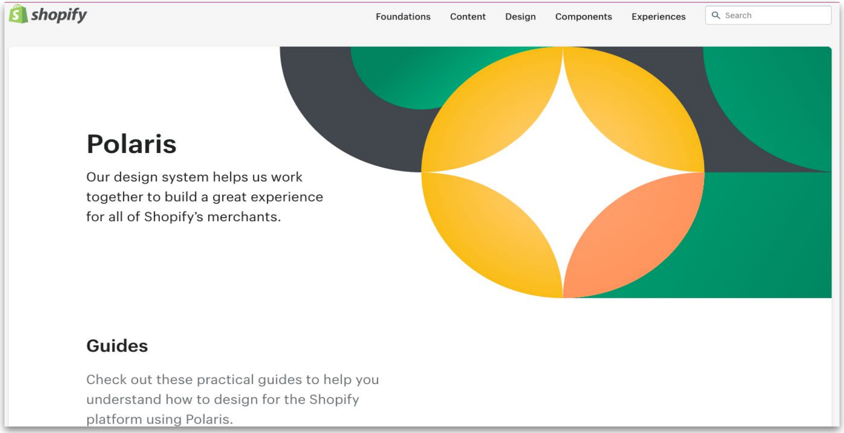
Landing page - What worked

Tasks: all

✓ Clear landing pages with fewer options to choose from



Foundations Content Design Components Experiences



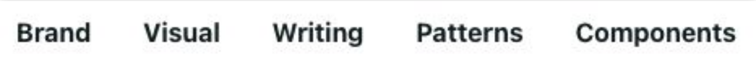
Polaris

Our design system helps us work together to build a great experience for all of Shopify's merchants.

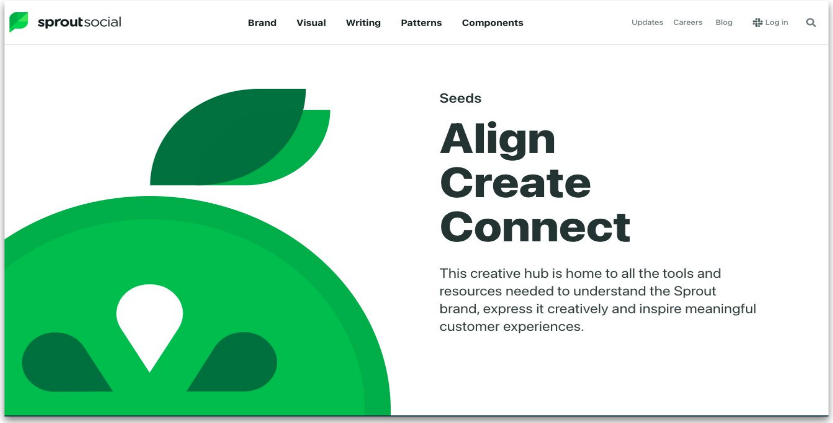
Guides

Check out these practical guides to help you understand how to design for the Shopify platform using Polaris.

Shopify



Brand Visual Writing Patterns Components



Seeds

Align
Create
Connect

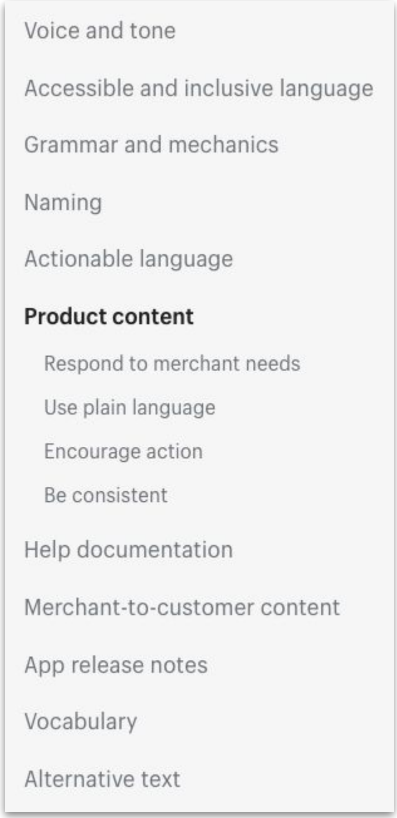
This creative hub is home to all the tools and resources needed to understand the Sprout brand, express it creatively and inspire meaningful customer experiences.

Sprout

Global navigation - What didn’t work

Tasks: all

✕ Not landing at the beginning of a section: on Shopify, clicking on “Content” on the landing page (not the menu) lands you in the middle (Product content) of the Content section (not at the top).



Voice and tone

Accessible and inclusive language

Grammar and mechanics

Naming

Actionable language

Product content

- Respond to merchant needs
- Use plain language
- Encourage action
- Be consistent

Help documentation

Merchant-to-customer content

App release notes

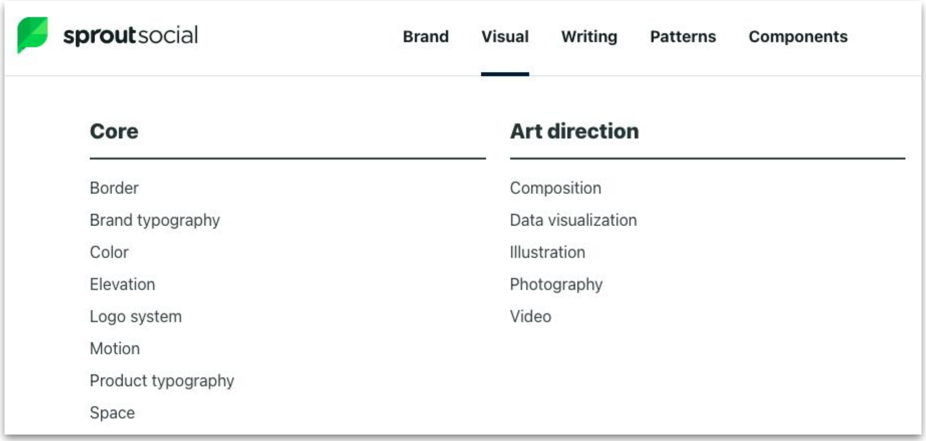
Vocabulary

Alternative text

Global navigation - What worked

Tasks: all

✓ The ability to expand the menu items and explore the options before committing was useful. Video



sproutsocial

Brand Visual Writing Patterns Components

Core

- Border
- Brand typography
- Color
- Elevation
- Logo system
- Motion
- Product typography
- Space

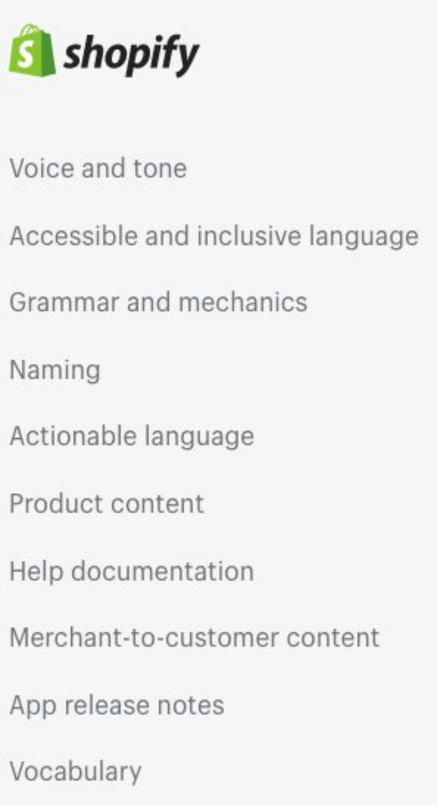
Art direction

- Composition
- Data visualization
- Illustration
- Photography
- Video

In-section navigation - What didn’t work

Tasks: all

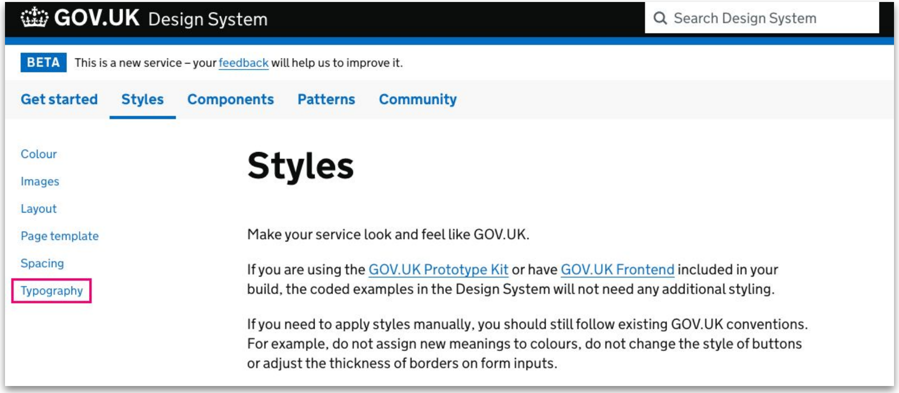
- ✗ Local menus with competing headings in Shopify. Using Ctrl-F to search within multiple pages Video
- ✗ Clicking on an item (Title and text) and landing on first sub-items (Captions) Video



In-section navigation - What worked

Tasks: all

✓ Being able to see the elements of a section in the left menu - like Typography in GOV.UK or Buttons in USWDS



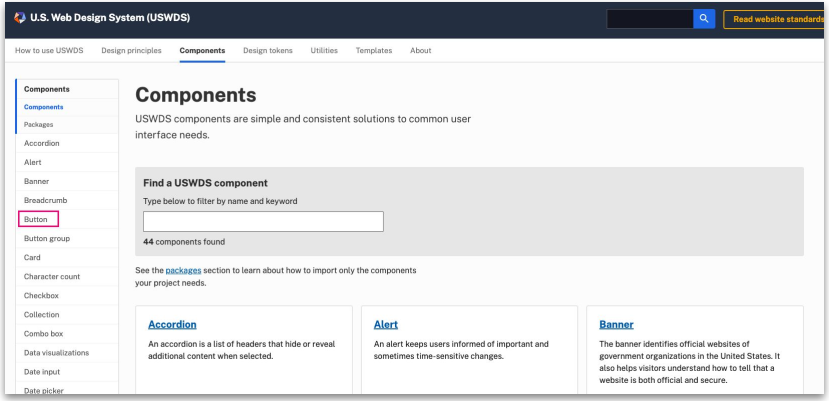
This is a new service – your feedback will help us to improve it.

Styles

Make your service look and feel like GOV.UK.

If you are using the GOV.UK Prototype Kit or have GOV.UK Frontend included in your build, the coded examples in the Design System will not need any additional styling.

If you need to apply styles manually, you should still follow existing GOV.UK conventions. For example, do not assign new meanings to colours, do not change the style of buttons or adjust the thickness of borders on form inputs.



Components

USWDS components are simple and consistent solutions to common user interface needs.

Find a USWDS component

Type below to filter by name and keyword

44 components found

See the packages section to learn about how to import only the components your project needs.

Accordion

An accordion is a list of headers that hide or reveal additional content when selected.

Alert

An alert keeps users informed of important and sometimes time-sensitive changes.

Banner

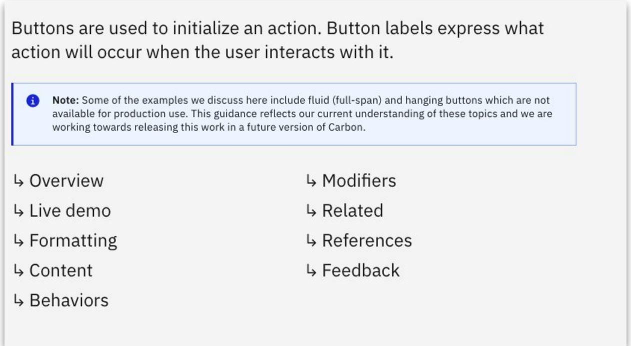
The banner identifies official websites of government organizations in the United States. It also helps visitors understand how to tell that a website is both official and secure.

On-page navigation - What didn’t work

Tasks: all

✕ Carbon’s on-page nav format that doesn’t look like links - most participants skipped over it. It doesn’t help a long and poorly organized page

✕ Right-hand menus. Were not used and created more noise (Salesforce, Atlassian). Some participants didn’t understand the right-hand menu is page table of contents, and scrolled down instead. Video



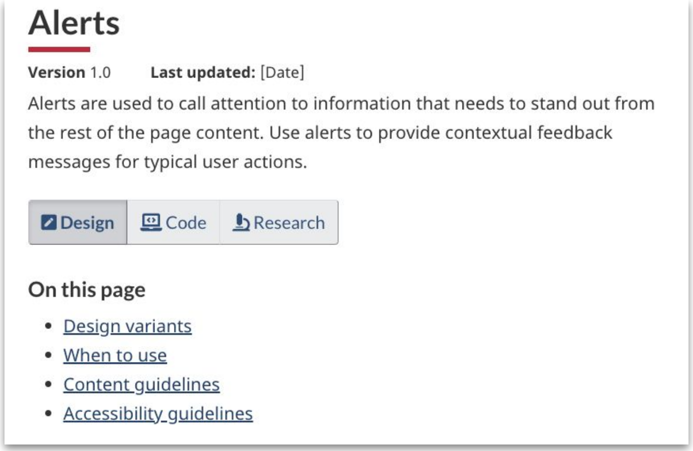
Buttons are used to initialize an action. Button labels express what action will occur when the user interacts with it.

Note: Some of the examples we discuss here include fluid (full-span) and hanging buttons which are not available for production use. This guidance reflects our current understanding of these topics and we are working towards releasing this work in a future version of Carbon.

On-page navigation - What worked

Tasks: all

- ✓ Borealis tabs to get to Code
- ✓ Clone tabs at the top (for changelog)



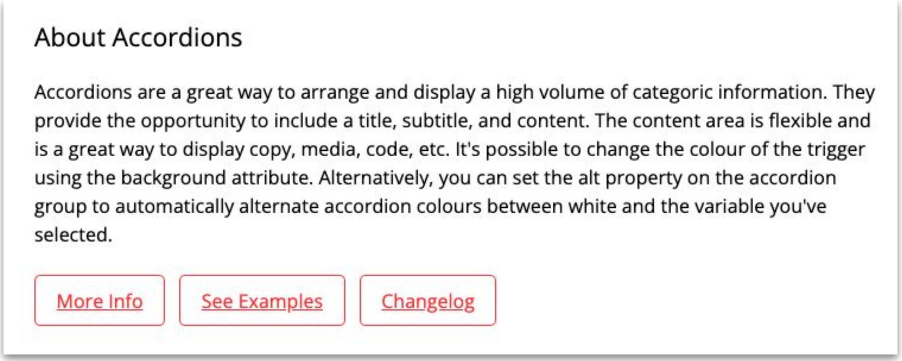
Alerts

Version 1.0 **Last updated:** [Date]

Alerts are used to call attention to information that needs to stand out from the rest of the page content. Use alerts to provide contextual feedback messages for typical user actions.

On this page

- Design variants
- When to use
- Content guidelines
- Accessibility guidelines



About Accordions

Accordions are a great way to arrange and display a high volume of categoric information. They provide the opportunity to include a title, subtitle, and content. The content area is flexible and is a great way to display copy, media, code, etc. It's possible to change the colour of the trigger using the background attribute. Alternatively, you can set the alt property on the accordion group to automatically alternate accordion colours between white and the variable you've selected.

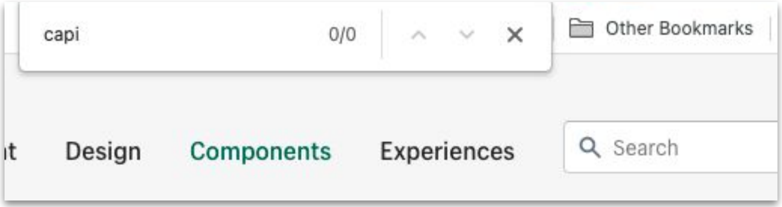
Search - What didn’t work

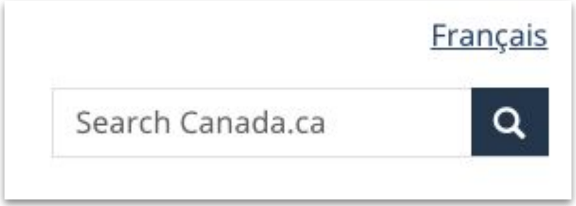
Tasks: all

- ✕ Local search on the right was often missed (Shopify, Atlassian, Gov.uk). Some participants were afraid it would be a global search and some people used Ctrl-F instead. Video

- ✕ No search results with appropriate keywords (Capitalization on Atlassian)

- ✕ Lack of good metadata led to unsuccessful search when people didn’t use the “right” keyword. Video

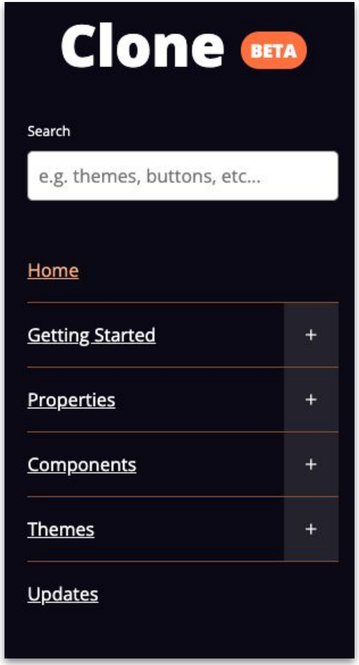

capi 0/0



Search - What worked

Tasks: all

- ✓ The filtered table on Canada.ca allowed quick findability of components. People are sure it’s not a global site-wide search
- ✓ Prominent local search on the left (Clone and Lighting) allowed quick findability. [Video]



Heading labels - What didn’t work

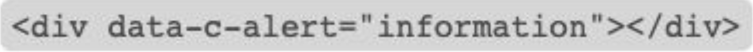
Tasks: all

✕ Having both users and elements on the landing page. Developers need components. (Carbon, Aurora)

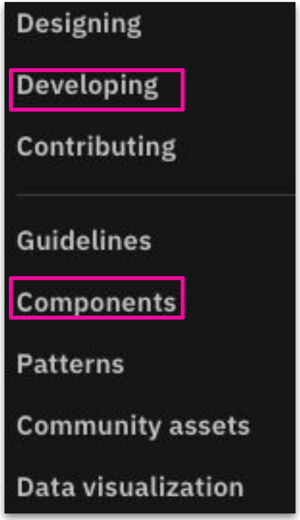
✕ Labels on the left menu that were not organized alphabetically (NZ).

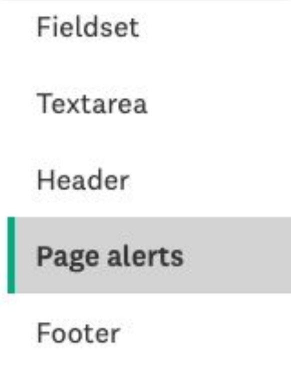
✕ Labels without the standard name: Page alerts instead of alerts (NZ).

✕ Labels and headings shouldn’t be code (Clone)



```
<div data-c-alert="information"></div>
```



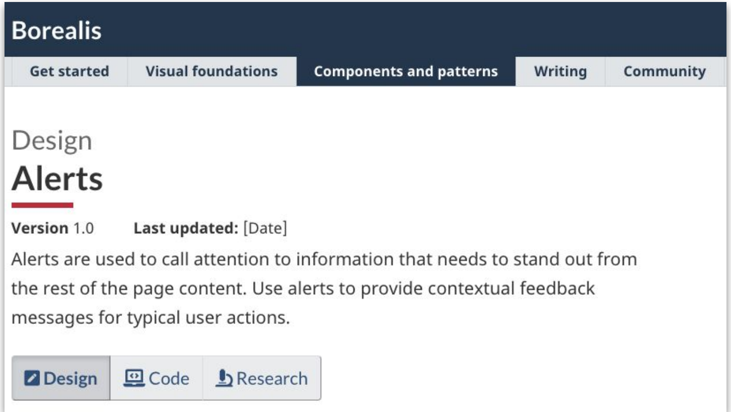


Heading labels - What worked

Tasks: all

- ✓ For finding the alert component, most participants first clicked on “Components” (30/35) - Components works as a label

- ✓ “Get started” worked well for the installation task.



Design

Alerts

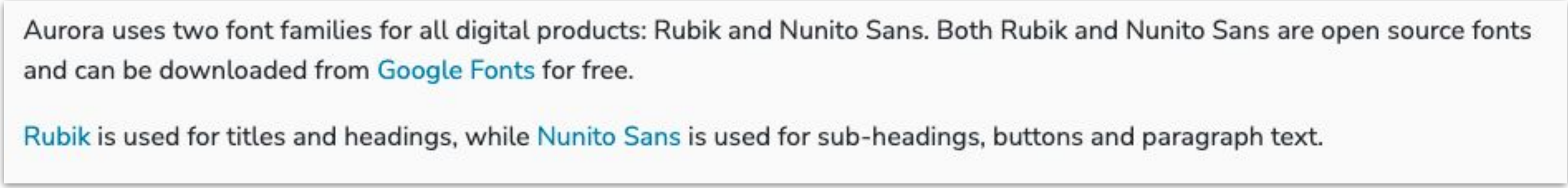
Version 1.0 **Last updated:** [Date]

Alerts are used to call attention to information that needs to stand out from the rest of the page content. Use alerts to provide contextual feedback messages for typical user actions.

Visual foundations - What didn’t work

Tasks: Find font and font size. Find primary colour

- ✗ Lack of actual color samples in Canada.ca didn’t help. Video
- ✗ Long and busy page on USWDS made it hard to follow.
- ✗ Font and font size not together, not prominent or surrounded by code and many visual examples on GOV.UK and Aurora. Video



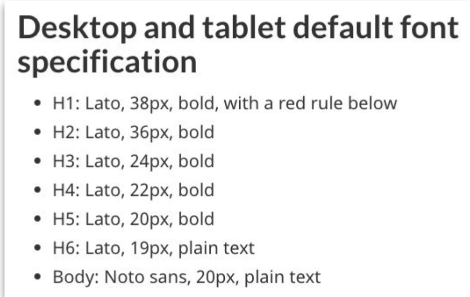
Aurora uses two font families for all digital products: Rubik and Nunito Sans. Both Rubik and Nunito Sans are open source fonts and can be downloaded from Google Fonts for free.

Rubik is used for titles and headings, while Nunito Sans is used for sub-headings, buttons and paragraph text.

Visual foundations - What worked

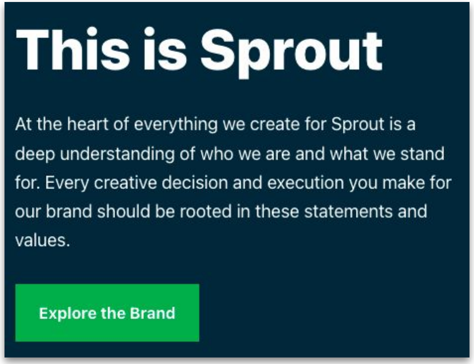
Tasks: Find font and font size. Find primary colour

- ✓ The expandable menu in Sprout worked well to get to the Color section quickly
- ✓ Clear specifications on fonts and fonts size on Canada.ca
- ✓ The design system itself as an explicit example of components



Desktop and tablet default font specification

- H1: Lato, 38px, bold, with a red rule below
- H2: Lato, 36px, bold
- H3: Lato, 24px, bold
- H4: Lato, 22px, bold
- H5: Lato, 20px, bold
- H6: Lato, 19px, plain text
- Body: Noto sans, 20px, plain text



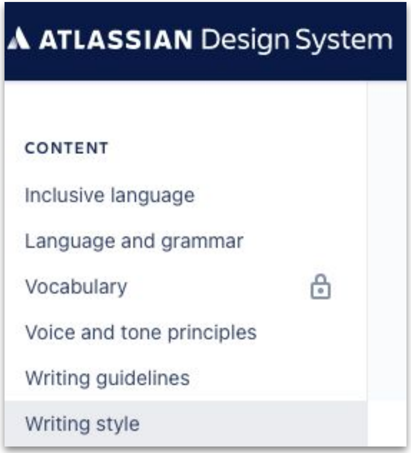
This is Sprout

At the heart of everything we create for Sprout is a deep understanding of who we are and what we stand for. Every creative decision and execution you make for our brand should be rooted in these statements and values.

General content guidance - What didn’t work

Task: How to capitalize headings

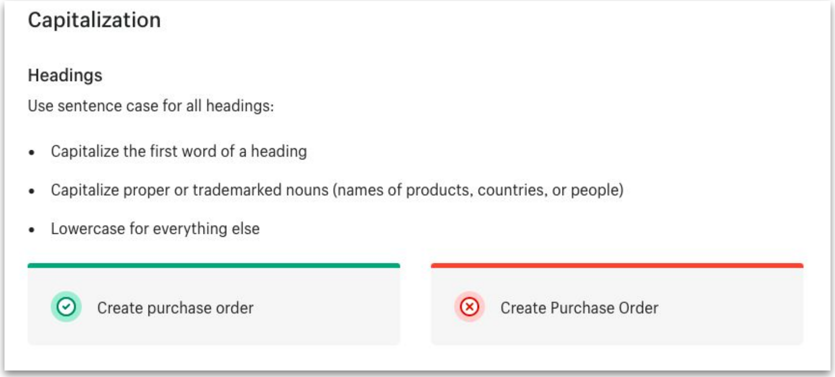
- ✗ Hesitation on the landing page. Where to go? Content or Design?: “*Is content design visual or writing elements?*”
- ✗ Unclear labels, like “Grammar and mechanics” on Shopify
- ✗ Competing labels: Writing style vs. Language and grammar. Video
- ✗ Confusion between local search and Ctrl-F. Using Ctrl-F in the wrong sections. Video



General content guidance - What worked

Task: How to capitalize headings

- ✓ Style is a familiar heading in Canada.ca. [Video]
  - ○ Ctrl-F on style guide
- ✓ The guidance is also easily found from the landing page. [Video]
- ✓ Clear do's and don'ts on Shopify



Capitalization

Headings

Use sentence case for all headings:

- Capitalize the first word of a heading
- Capitalize proper or trademarked nouns (names of products, countries, or people)
- Lowercase for everything else

✓ Create purchase order

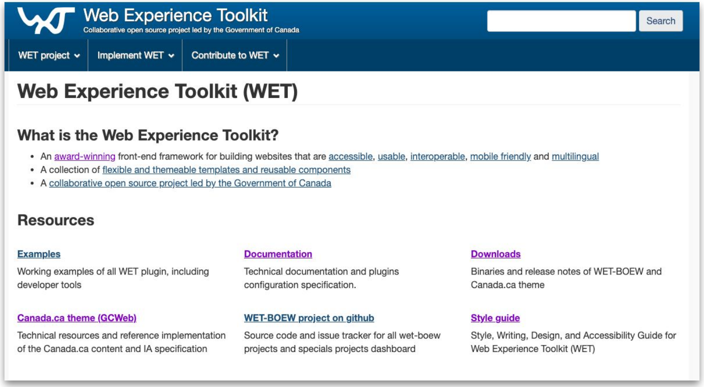
☒ Create Purchase Order

Installation - What didn’t work

Task: How to install the design system

✗ The separation of Downloads and Implementation guidelines in WET caused failure. Video

✗ Splitting of installation instructions in Carbon caused issues - a participant found partial instructions and assumed he had everything. Video



Web Experience Toolkit (WET)

What is the Web Experience Toolkit?

- An award-winning front-end framework for building websites that are accessible, usable, interoperable, mobile friendly and multilingual
- A collection of flexible and themeable templates and reusable components
- A collaborative open source project led by the Government of Canada

Resources

Examples
Working examples of all WET plugin, including developer tools

Documentation
Technical documentation and plugins configuration specification.

Downloads
Binaries and release notes of WET-BOEW and Canada.ca theme

Canada.ca theme (GCWeb)
Technical resources and reference implementation of the Canada.ca content and IA specification

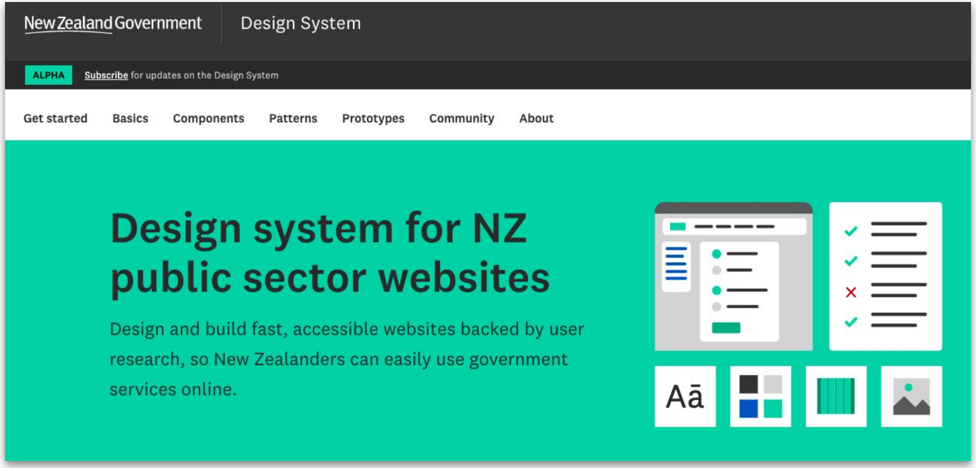
WET-BOEW project on github
Source code and issue tracker for all wet-boew projects and specials projects dashboard

Style guide
Style, Writing, Design, and Accessibility Guide for Web Experience Toolkit (WET)

Installation - What worked

Task: How to install the design system

- ✓ A majority (9/15) used “Get started” in the menu (7/10 on Carbon and NZ)
- ✓ Instructions with npm package manager worked well for some developers (not for *loyal implementers*) . Video.



Design system for NZ public sector websites

Design and build fast, accessible websites backed by user research, so New Zealanders can easily use government services online.

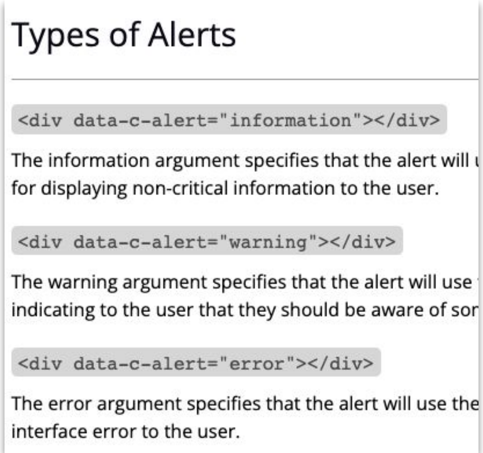
Component guidance findings

Design
Content
Code
Versioning

Component design guidance - What didn’t work

Task: Choosing the right type of alert

- ✗ NZ’s Alert component page was too long and wordy. Long introduction to a component. [Video]
- ✗ Mixing types of guidances (Code, content)
- ✗ No clear headings and subheading. Participants had problems distinguishing between alert variants on Clone



Types of Alerts

```
<div data-c-alert="information"></div>
```

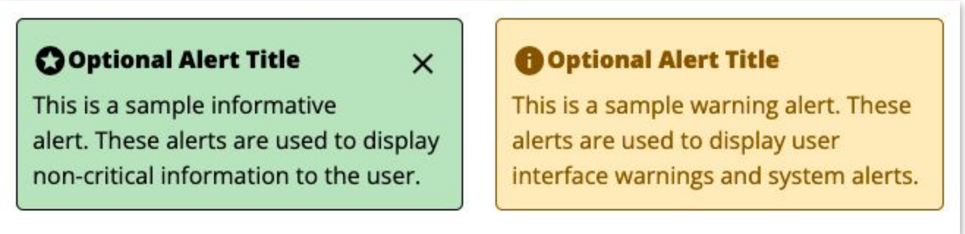
The information argument specifies that the alert will
for displaying non-critical information to the user.

```
<div data-c-alert="warning"></div>
```

The warning argument specifies that the alert will use
indicating to the user that they should be aware of sor

```
<div data-c-alert="error"></div>
```

The error argument specifies that the alert will use the
interface error to the user.



Optional Alert Title ✕

This is a sample informative alert. These alerts are used to display non-critical information to the user.

Optional Alert Title

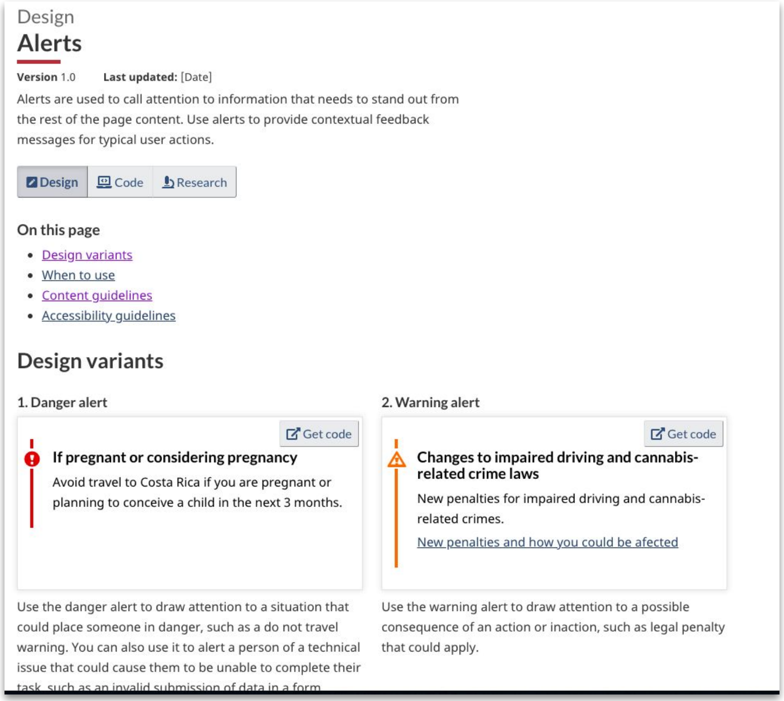
This is a sample warning alert. These alerts are used to display user interface warnings and system alerts.

Component design guidance - What worked

Task: Choosing the right type of alert

- ✓ Concise explanation of the component with easy access to all its aspects (design, guidance, code and versioning)

- ✓ Having the variants and guidance first on the component page worked very well in Borealis. Video



Design

Alerts

Version 1.0 **Last updated:** [Date]

Alerts are used to call attention to information that needs to stand out from the rest of the page content. Use alerts to provide contextual feedback messages for typical user actions.

On this page

- Design variants
- When to use
- Content guidelines
- Accessibility guidelines

Design variants

1. Danger alert

If pregnant or considering pregnancy

Avoid travel to Costa Rica if you are pregnant or planning to conceive a child in the next 3 months.

Use the danger alert to draw attention to a situation that could place someone in danger, such as a do not travel warning. You can also use it to alert a person of a technical issue that could cause them to be unable to complete their task, such as an invalid submission of data in a form.

2. Warning alert

Changes to impaired driving and cannabis-related crime laws

New penalties for impaired driving and cannabis-related crimes.

New penalties and how you could be afected

Use the warning alert to draw attention to a possible consequence of an action or inaction, such as legal penalty that could apply.

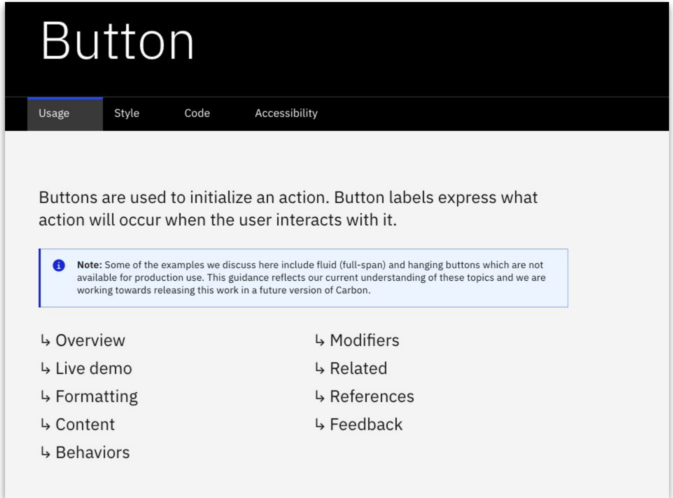
Component content guidance - What didn’t work

Task: How to write good button text

- ✗ Participants were not able to derive content guidance from examples in the Canada.ca design system - needs an explicit section
  “It’s not clear… I mean it’s there, but… How to write content for a button…”

- ✗ Looking into “general writing guidance” and never got to the buttons guidance - needs a rescue for component content guidance

- ✗ Carbon’s very long component guidance page didn’t help people. And most participants didn’t see the table of contents ot tabs at the top. Video



Button

Buttons are used to initialize an action. Button labels express what action will occur when the user interacts with it.

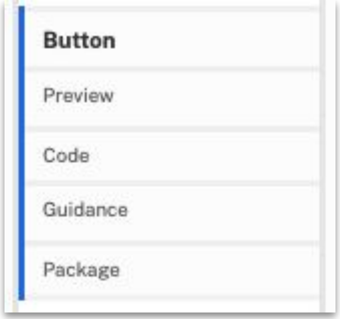
Note: Some of the examples we discuss here include fluid (full-span) and hanging buttons which are not available for production use. This guidance reflects our current understanding of these topics and we are working towards releasing this work in a future version of Carbon.

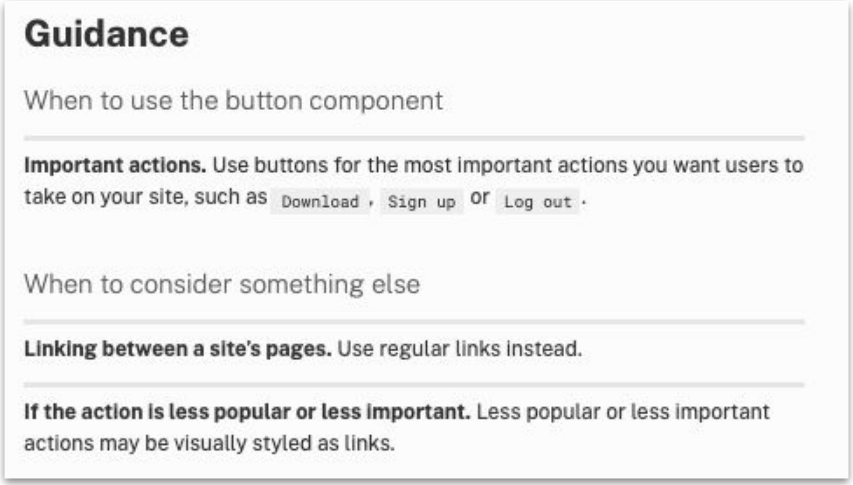
↳ Overview
↳ Live demo
↳ Formatting
↳ Content
↳ Behaviors
↳ Modifiers
↳ Related
↳ References
↳ Feedback

Component content guidance - What worked

Task: How to write good button text

- ✓ Short, to-the-point content guidance in a clear separated section
- ✓ Clear content guidelines in the Buttons page in USWDS (Formatting could be less confusing). Video





Guidance

When to use the button component

Important actions. Use buttons for the most important actions you want users to take on your site, such as `Download`, `Sign up` or `Log out`.

When to consider something else

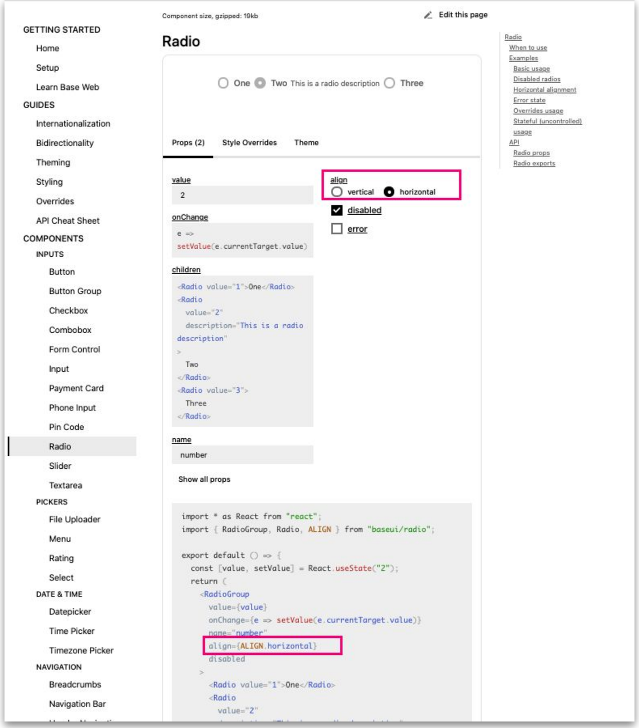
Linking between a site's pages. Use regular links instead.

If the action is less popular or less important. Less popular or less important actions may be visually styled as links.

Component code guidance- What didn’t work

Task: Class to use to make radio buttons horizontal

- ✕ 1 participant on GOV.UK was attracted by Styles (thinking it would be a Coding Style guide). Video
- ✕ On Uber, it wasn’t easy to make the connection between ticking “horizontal” and finding the code it had changed. Video



Radio

○ One ● Two This is a radio description ○ Three

value: 2

onChange:

```
e =>
setValue(e.currentTarget.value)
```

children:

```
<Radio value="1">One</Radio>
<Radio
  value="2"
  description="This is a radio
description"
>
  Two
</Radio>
<Radio value="3">
  Three
</Radio>
```

name: number

align: ○ vertical ● horizontal

☑ disabled

□ error

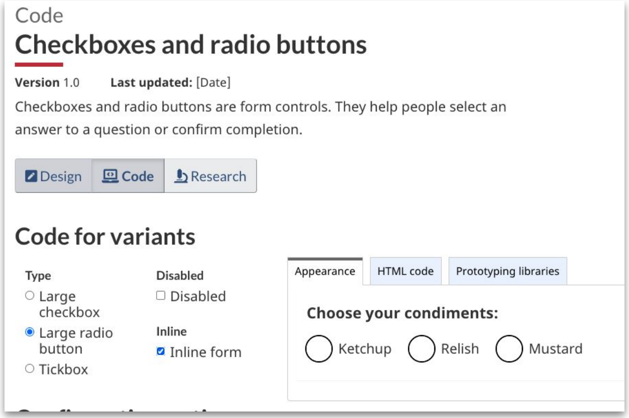
```
import * as React from "react";
import { RadioGroup, Radio, ALIGN } from "baseui/radio";

export default () => {
  const [value, setValue] = React.useState("2");
  return (
    <RadioGroup
      value={value}
      onChange={e => setValue(e.currentTarget.value)}
      name="number"
      align={ALIGN.horizontal}
      disabled
    >
      <Radio value="1">One</Radio>
      <Radio
        value="2"
```

Component code guidance - What worked

Task: Class to use to make radio buttons horizontal

- ✓ All 5 participants got to the Code tab in Borealis (3 directly on the tab, 2 by clicking on Get code on variants).
- ✓ Interactive options that show visual and code worked well - maybe would be better if you could see visual and code together. [Video]



Code

Checkboxes and radio buttons

Version 1.0 **Last updated:** [Date]

Checkboxes and radio buttons are form controls. They help people select an answer to a question or confirm completion.

Code for variants

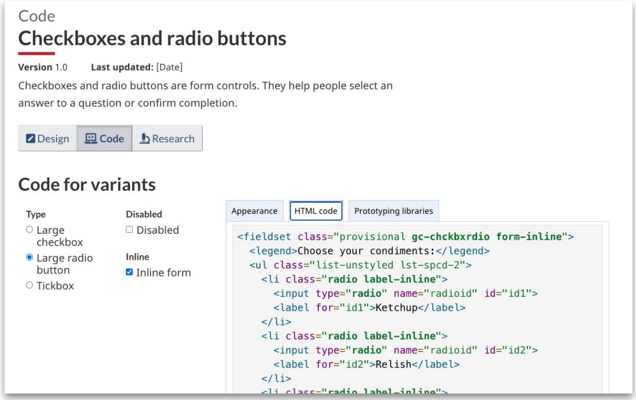
Type
○ Large checkbox
● Large radio button
○ Tickbox

Disabled
□ Disabled

Inline
☑ Inline form

Choose your condiments:

○ Ketchup ○ Relish ○ Mustard



Code

Checkboxes and radio buttons

Version 1.0 **Last updated:** [Date]

Checkboxes and radio buttons are form controls. They help people select an answer to a question or confirm completion.

Code for variants

Type
○ Large checkbox
● Large radio button
○ Tickbox

Disabled
□ Disabled

Inline
☑ Inline form

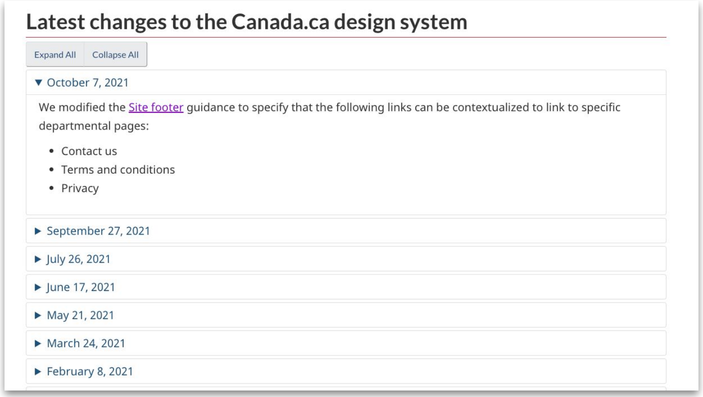
```
<fieldset class="provisional gc-chckbxrdio form-inline">
  <legend>Choose your condiments:</legend>
  <ul class="list-unstyled lst-spcd-2">
    <li class="radio label-inline">
      <input type="radio" name="radioid" id="id1">
      <label for="id1">Ketchup</label>
    </li>
    <li class="radio label-inline">
      <input type="radio" name="radioid" id="id2">
      <label for="id2">Relish</label>
    </li>
    <li class="radio label-inline">
```

Component versioning - What didn’t work

Task: Identify the latest change to the button component

- ✗ The need to expand all the previous changes on Canada.ca caused failure. Video

- ✗ Having to scroll down a lengthy Button page to find latest changes (Lightning) didn’t work. Video



Latest changes to the Canada.ca design system

▼ October 7, 2021

We modified the Site footer guidance to specify that the following links can be contextualized to link to specific departmental pages:

- Contact us
- Terms and conditions
- Privacy

▶ September 27, 2021

▶ July 26, 2021

▶ June 17, 2021

▶ May 21, 2021

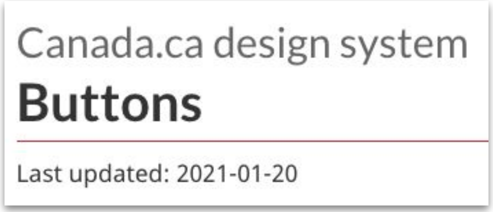
▶ March 24, 2021

▶ February 8, 2021

Component versioning - What worked

Task: Identify the latest change to the button component

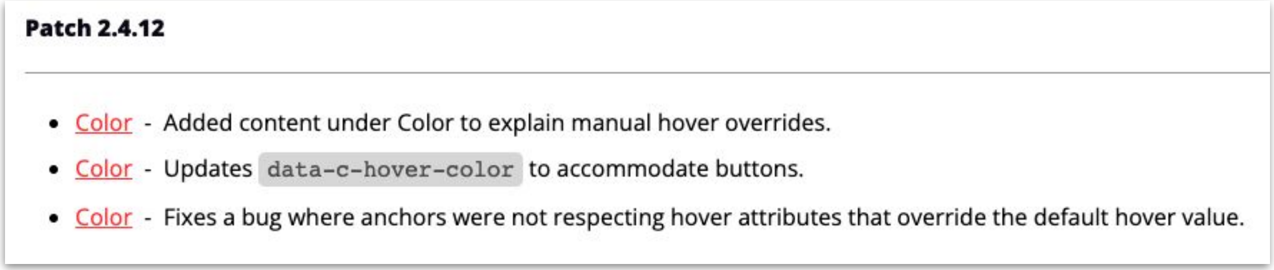
- ✓ 8/15 trials started with an “overall change” link on the guidance page, and 7/15 started by going to the relevant component. It’s important to have both options
- ✓ A link at the top of the component page worked well for Clone and Canada.ca. [Video]()
- ✓ Using numbered release (semantic versioning) helped some participants



Canada.ca design system

Buttons

Last updated: 2021-01-20



Patch 2.4.12

- Color - Added content under Color to explain manual hover overrides.
- Color - Updates `data-c-hover-color` to accommodate buttons.
- Color - Fixes a bug where anchors were not respecting hover attributes that override the default hover value.

Users’ suggestions on the Canada.ca design system

Users’ suggestions for Canada.ca

Keep:

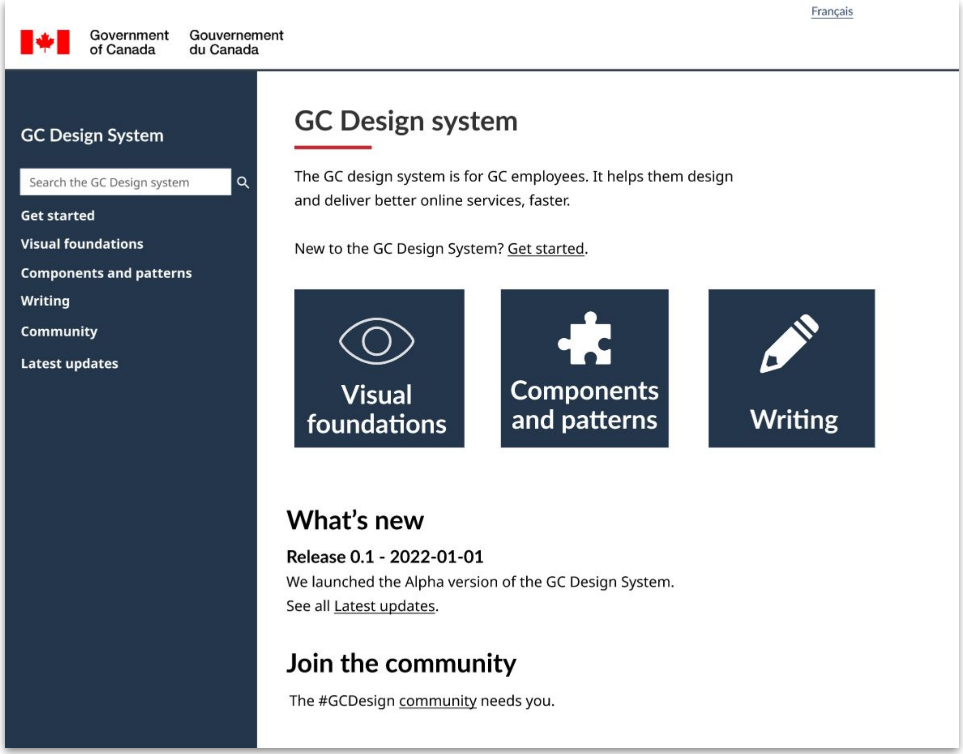
- Filter on the landing page Video
- Style guide
- Guidance with examples (it’s an authoritative source to know when to use, what to avoid).

Add:

- IA: “Taking what's already there and putting structure around it”. Video
- Integration of Style guide and patterns (bring in content guidance within the component guidance)
- Forms and tables, fieldflow, input
- Aesthetics
- Make clear what’s mandatory, what’s optional
- More specific guidance with customization

Starting point for an alpha prototype

Rough prototype



Français

Government of Canada Gouvernement du Canada

GC Design System

Search the GC Design system

- Get started
- Visual foundations
- Components and patterns
- Writing
- Community
- Latest updates

GC Design system

The GC design system is for GC employees. It helps them design and deliver better online services, faster.

New to the GC Design System? Get started.

Visual foundations

Components and patterns

Writing

What's new

Release 0.1 - 2022-01-01

We launched the Alpha version of the GC Design System.

See all Latest updates.

Join the community

The #GCDesign community needs you.

Figma prototype

Important note: the purpose of the Figma prototype is not to propose a specific design; it's to illustrate some of the recommendations that stem from the findings.

Evidence-informed starting points

IA and navigation

- Left nav where you could see all sections and items in the section
- Limited number of distinct main sections.
- Initial proposal:
    - Get started, Visual foundations, Components and patterns, Writing, Community, and Latest updates
- Do additional research on the naming of the main sections
- The left navigation needs to be part of the design system itself as a component
- Needs to work on a mobile version of that menu
- More linkages between sections (between pattern and style guide, for example) - when things can be applicable. (E.g. how to write text for a buttons: you'd think it would be covered in the style guide - needs a cross link)

Evidence-informed starting points

Search

- Have system-wide search that functions as a filter (not SERP)
- Include the search in the left menu and on top, without the need to scroll down
- Have a good set of metadata on items so things come up with different queries: plural, synonymous, misspelled words (i.e. color, colour, typography, font, radio, ratio)
- Search all items in the system at once (not just patterns and templates)

Evidence-informed starting points

Visual foundations

- Have a separate section for design tokens and visual foundations (like colours, typography, etc.)
- Start by naming it something like Visual foundations (not Styles or Design)
- More research is needed to make sure the label isn’t confusing
- Have guidance, visual examples, tokens and code together, but in a way that isn’t confusing - more research and design is needed (a Jobs-to-be-done exercise to understand how people would use these elements could help)
- Provide elements in a design kit (e.g. Figma), so as not to have to reverse-engineer

Evidence-informed starting points

Content guidance

- General content guidance rules in a separate section.
- Start with something like Writing or Content
- More research is needed for the label
- Include specific “content guidelines” in relevant components
- Include a rescue to component-specific guidance in the General content guidance section

Evidence-informed starting points

Component guidance

- Keep guidance short and to the point - no super long pages
- Have usage and code in the same page, but in separate sections
- For code, expand collapses, tabs or a code section at the bottom of the page are alternatives to be tested
- Develop something similar to what Borealis has (borrowed from the CRA web service manual) to tick options and variants and see code changes easily
- Include a “Latest changes” or “Version history” link at the top of each component

Evidence-informed starting points

Installation and usage guidance

- Include a “Get started” link in the main menu
- Include installation instructions and onboarding documentation under Get started
- Use a modern npm package manager and clear installation instructions, in one spot - will need proper training for employees less familiar with this option

Meeting the needs of the different users

> *"The thing I really love about working on a design system is that I'm reconciling two complex systems—design and code—both of which use their own languages and have speakers or practitioners of those languages who use different words for the same concepts."* - Why do design systems need content strategists?

- Some design systems are design-oriented, some are dev-oriented
- Code can be intimidating for non-coders
- How might we build a component page that covers **design**, **code**, and **content** in a way that works for:
    - Communicators
    - Designers
    - Coders and publishers

Next steps

- **Watch clips** from the testing
- Conduct a **Jobs-to-be-done** exercise to better understand tasks people will try to accomplish with the documentation site
- Do a **first-click research** to test the proposed main sections
- **Start building** the alpha documentation site using findings from this deck as a starting point
- Include a **few key components** in the prototype
- Do another round of **usability testing with the alpha documentation site** - adjust as needed

Annexes

Methodology - Tasks

Communicators and designers test

Topic	Task	Design System
General content guidance (Writing)	You and your colleague are arguing about how to capitalize headings on a web page. Should you capitalize only the first word of a heading, or all words of a heading?	L1 - Caps - Shopify L2 - Caps - Canada.ca L3 - Caps - Atlassian
Component-specific content guidance (Buttons)	You are working on a page and you're adding a button to launch a service. You want to write the best possible call to action. Name a few things you should consider to write good text for a button.	B1 - Buttons - Carbon B2 - Buttons - Canada.ca B3 - Buttons - USWDS
General design guidance (Fonts)	You're starting to design a new online service. Which font and font size should you use for the body text?	F1 - Font - GOV.UK F2 - Font - Aurora F3 - Font - Canada.ca
Specific component design guidance (Alerts)	You want to add an alert on a page to bring to attention something that users need to be aware of to avoid a problem or negative consequences. Which type of alert should you use?	A1 - Alert - New Zealand A2 - Alert - Clone A3 - Alert - Borealis

Methodology - Tasks

Developers test

Topic	Task	Design System
Component- code guidance (Class)	You need to display a set of radio buttons horizontally, instead of vertically. Which attribute or class would you use?	C1 - Class - Borealis (proto) C2 - Class - GOV.UK C3 - Class - Uber
Visual foundations (Design tokens)	You've been asked to build a new component for a service. Name the variable/token you should use for the primary colour.	D1 - Design tokens - Sprout D2 - Design tokens - Canada.ca D3 - Design tokens - USWDS
Installing (Install)	This is your first day in your new developer job. How do you install the latest version of the design system to start working?	I1 - Install - Carbon I2 - Install - New Zealand I3 - Install - WET
Versioning (Change history)	Your client tells you something has recently changed with buttons in the design system. Find out what's the latest change in the Buttons component.	V1 - Change - Lightning V2 - Change - Clone V3 - Change - Canada.ca

Annex - Capitalization

Scenario:

You and your colleague are arguing about how to capitalize headings on a web page. Should you capitalize only the first word of a heading, or all words of a heading?

	Shopify	Canada.ca	Atlassian
Findability	10/12	**11/11**	8/12
Time to findability	75.1	**34.8**	72.0
Success	10/12	**11/11**	7/12
Time to success	88.9	**51.0**	75.3

How participants ranked the design systems and why:

1. **Canada.ca: Familiarity. Clear and easy to find guidance**
2. Shopify: Nice layout. Headings were confusing.
3. Atlassian: Competing headings, small font, poor search results

Annex - Buttons

Scenario:

You are working on a page and you're adding a button to launch a service. You want to write the best possible call to action. Name a few things you should consider to write good text for a button.

	Carbon (IBM)	Canada.ca	USWDS
Findability	9/12	10/12	**11/12**
Time to findability	47.4	17.4	**40.5**
Success	3/12	0/12	**10/12**
Time to success	127.0	0.0	**89.5**

How participants ranked the design systems and why:

1. **USWDS: Explicit guidance. Design, content guidance and code on the same page**
2. Canada.ca: Good guidance but not for label buttons
3. Carbon: Overwhelming. Too much information not relevant for the taks

Annex - Fonts

Scenario:

You're starting to design a new online service. Which font and font size should you use for the body text?

	GOV.UK	Aurora	Canada.ca
Findability	**11/11**	10/11	10/11
Time to findability	**38.4**	52.1	57.0
Success	8/11	9/11	**10/11**
Time to success	88.8	126.7	**77.5**

How participants ranked the design systems:

1. **Canada.ca: filterable table worked well (could type font and find typography). Clear answer where font and font size are together**
2. GOV.UK: Easy to find, but tough to understand
3. Aurora: Fonts under Components. Confusing guidance surrounded by visual examples

Annex - Alerts

Scenario:

You want to add an alert on a page to bring to attention something that users need to be aware of to avoid a problem or negative consequences. Which type of alert should you use?

	New Zealand	Clone	Borealis (proto)
Findability	12/12	12/12	**12/12**
Time to findability	30.8	33.8	**26.3**
Success	8/12	7/12	**8/12**
Time to success	91.6	121.0	**83.1**

How participants ranked the design systems:

1. **Borealis (proto): Search worked well. Easy to understand different types of alerts**
2. Clone (Talent Cloud): headings were code, types of alerts are not easy to understand
3. New Zealand Government: Bad label (Page alert). Very long introduction to the component

Annex - Class to modify a component

Scenario:

You need to display a set of radio buttons horizontally, instead of vertically. Which attribute or class would you use?

	Borealis	**GOV.UK**	**Uber**
Findability	**5/5**	3/5	3/5
Time to findability	**81.6**	22.3	34.7
Success	**5/5**	3/5	3/5
Time to success	**137.8**	62.0	72.3

How participants ranked the design systems:

1. **Borealis: Easier to find. Easy to see the changes in the code**
2. Uber: Difficult to see the changes in the code
3. GOV.UK: More information between example and code so was difficult to spot

Annex - Design tokens

Scenario:

You've been asked to build a new component for a service. Name the variable/token you should use for the primary colour.

	Sprout	Canada.ca	USWDS
Findability	3/5	**4/5**	3/5
Time to findability	**12.7**	46.3	24.7
Success	3/5	2/5	**3/5**
Time to success	79.7	104.0	**42.3**

How participants ranked the design systems:

1. **Sprout: Visual examples and a good implementation on the design system**
2. USWDS: Too many colours, too many options
3. Canada.ca: No visual examples

Annex - Installing

Scenario:

This is your first day in your new developer job. How do you install the latest version of the design system to start working?

	Carbon	New Zealand	WET
Findability	3/5	**4/5**	3/5
Time to findability	38.3	**18.3**	38.3
Success	3/5	**3/5**	3/5
Time to success	60.7	**37.7**	81.7

How participants ranked the design systems:

1. **Carbon: Intuitive navigation (under Getting started). Clear tutorial**
2. WET: Installation and downloads are not together. Easy only for familiar users
3. New Zealand: Easy to arrive to the right page but more variants to understand

Annex - Change history

Scenario:

Your client tells you something has recently changed with buttons in the design system. Find out what's the latest change in the Buttons component.

	Lightning	Clone	Canada.ca
Findability	5/5	**5/5**	5/5
Time to findability	10.4	**13.0**	21.4
Success	3/5	**5/5**	5/5
Time to success	58.3	**34.6**	60.4

How participants ranked the design systems:

1. **Clone (Talent Cloud). On the left navigation. Organized by patch numbers. Also, clear on every component page**
2. Canada.ca. Easy to find but difficult to expand all the dates with changes.
3. Lightning (Salesforce): Had to scroll. Confusing right-hand navigation