PatternFly Community Meeting

May 5, 2020

www.patternfly.org/v4

Agenda

Topics

- New features and enhancements - Jenny Shandelman, Rebecca Alpert, and, Katie McFaul
- Colors and Css variables enhancements on site - Gina Doyle
- PatternFly content style guide - Abi Donahue
- E-training for Sketch library - Mary Shakshober
- Website research findings - Steve Cauffman

What's coming up

- What we're working on next - Rachael Petrie and Zack Allen

Q&A

New features and enhancements

- Card view - Jenny Shandelman
- Extra-large empty state - Rebecca Alpert
- Select enhancement- Katie McFaul

Topics

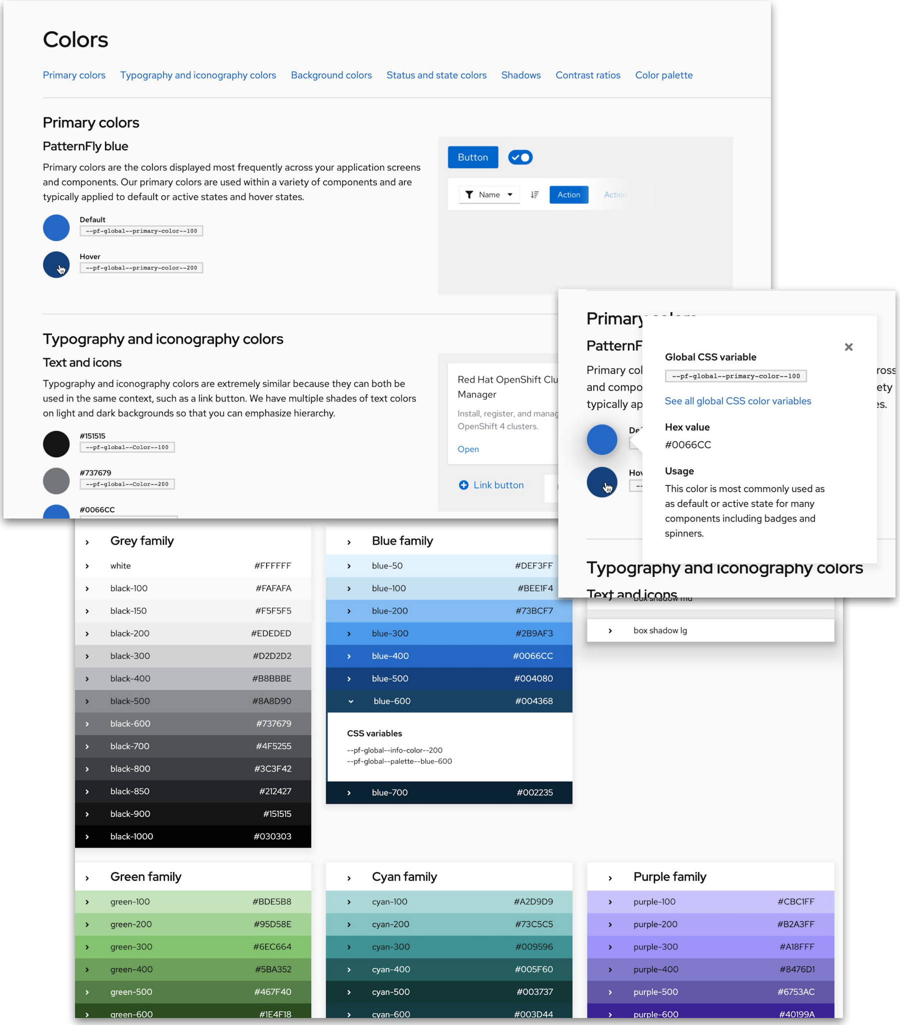
PatternFly site: Colors page and CSS variable mapping

Colors page See it live!

Updates and new additions

- Groupings of colors based on variables and how they are used within components
- Specifics per each color swatch including variable names, hex value, information about where that color is used
- See the whole color palette and for each color, get the palette color name, hex value and all variables associated with that color

P.S. Thank you Zack!



Colors

Primary colors | Typography and iconography colors | Background colors | Status and state colors | Shadows | Contrast ratios | Color palette

Primary colors

PatternFly blue

Primary colors are the colors displayed most frequently across your application screens and components. Our primary colors are used within a variety of components and are typically applied to default or active states and hover states.

Default `--pf-global--primary-color--100`

Hover `--pf-global--primary-color--200`

Typography and iconography colors

Text and icons

Typography and iconography colors are extremely similar because they can both be used in the same context, such as a link button. We have multiple shades of text colors on light and dark backgrounds so that you can emphasize hierarchy.

#151515 `--pf-global--Color--100`

#737679 `--pf-global--Color--200`

#0066CC

Red Hat OpenShift Clu
Manager

Install, register, and manag
OpenShift 4 clusters.

Global CSS variable

`--pf-global--primary-color--100`

See all global CSS color variables

Hex value

#0066CC

Usage

This color is most commonly used as as default or active state for many components including badges and spinners.

Typography and iconography colors

Text and icons

Grey family	
white	#FFFFFF
black-100	#FAFAFA
black-150	#F5F5F5
black-200	#EDEDED
black-300	#D2D2D2
black-400	#B8BBBE
black-500	#8A8D90
black-600	#737679
black-700	#4F5255
black-800	#3C3F42
black-850	#212427
black-900	#151515
black-1000	#030303

Blue family	
blue-50	#DEF3FF
blue-100	#BEE1F4
blue-200	#73BCF7
blue-300	#2B9AF3
blue-400	#0066CC
blue-500	#004080
blue-600	#004368

CSS variables

--pf-global--info-color--200
--pf-global--palette--blue-600

blue-700	#002235

box shadow md

box shadow lg

Green family	
green-100	#BDE5B8
green-200	#95D58E
green-300	#6EC664
green-400	#5BA352
green-500	#467F40
green-600	#1E4F18

Cyan family	
cyan-100	#A2D9D9
cyan-200	#73C5C5
cyan-300	#009596
cyan-400	#005F60
cyan-500	#003737
cyan-600	#003D44

Purple family	
purple-100	#CBC1FF
purple-200	#B2A3FF
purple-300	#A18FFF
purple-400	#8476D1
purple-500	#6753AC
purple-600	#40199A

Variable mapping

See it live!

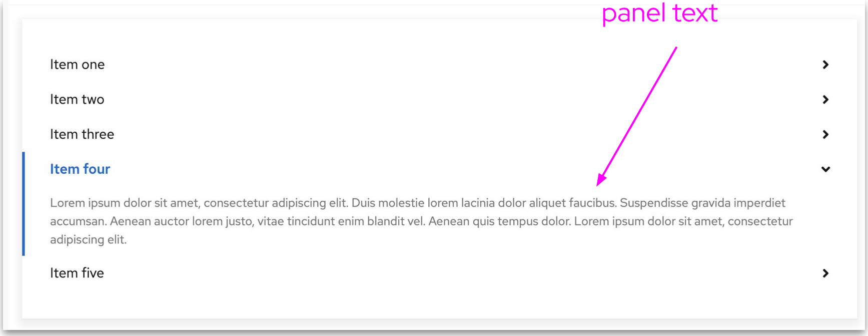
Updates and new additions

- CSS variables page is broken into 3 groups
    - CSS variables
    - Chart CSS variables
    - Font type CSS variables
- Variables per component show a full mapping of how that variable gets to the value (accordion example)

P.S. Thank you Joachim!

I want to know...

Color of expanded panel text



Item one

Item two

Item three

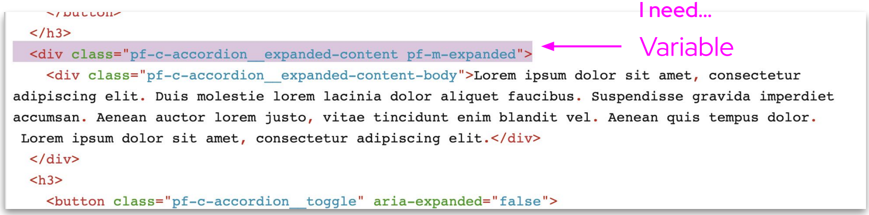
Item four

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis molestie lorem lacinia dolor aliquet faucibus. Suspendisse gravida imperdiet accumsan. Aenean auctor lorem justo, vitae tincidunt enim blandit vel. Aenean quis tempus dolor. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

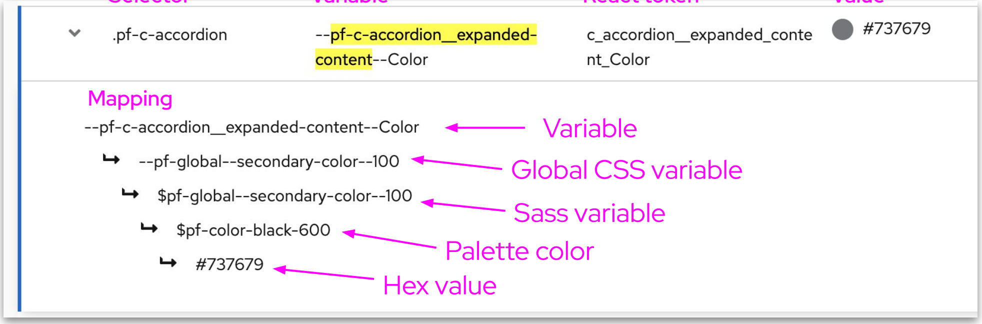
Item five

I need...

Variable



```
    </button>
  </h3>
  <div class="pf-c-accordion__expanded-content pf-m-expanded">
    <div class="pf-c-accordion__expanded-content-body">Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Duis molestie lorem lacinia dolor aliquet faucibus. Suspendisse gravida imperdiet
accumsan. Aenean auctor lorem justo, vitae tincidunt enim blandit vel. Aenean quis tempus dolor.
 Lorem ipsum dolor sit amet, consectetur adipiscing elit.</div>
  </div>
  <h3>
    <button class="pf-c-accordion__toggle" aria-expanded="false">
```



Selector	Variable	React token	Value
.pf-c-accordion	--pf-c-accordion__expanded-content--Color	c_accordion__expanded_content_Color	● #737679

Mapping

--pf-c-accordion__expanded-content--Color ← Variable

↳ --pf-global--secondary-color--100 ← Global CSS variable

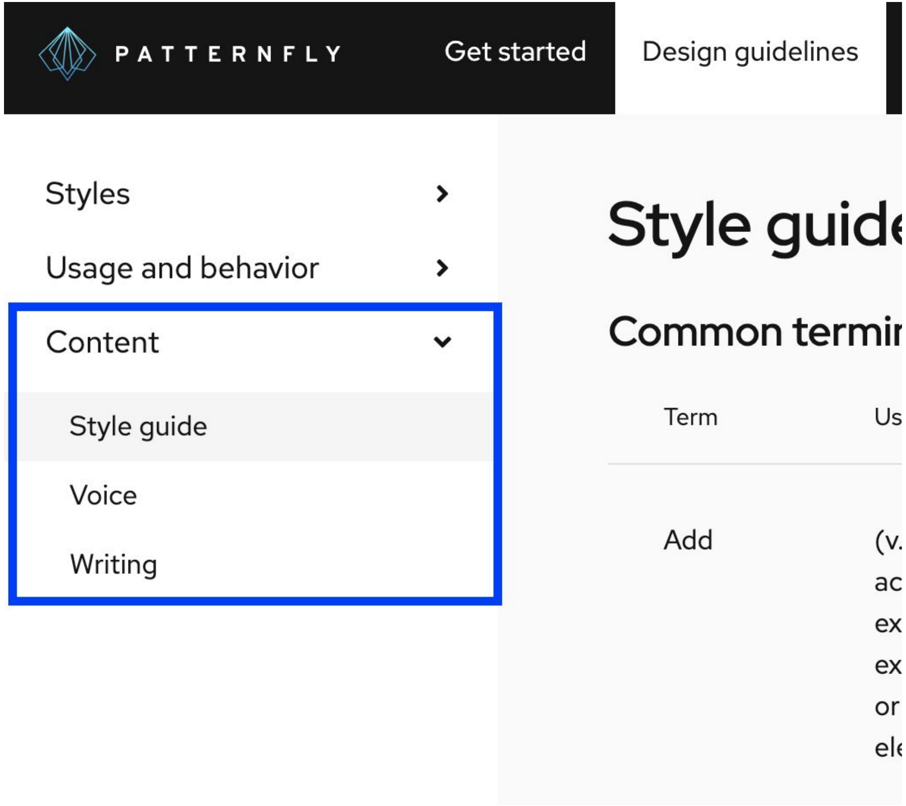
↳ $pf-global--secondary-color--100 ← Sass variable

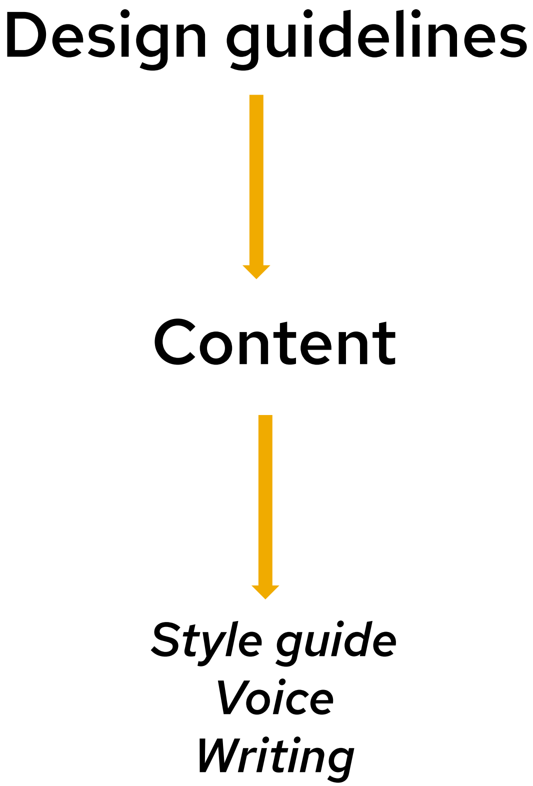
↳ $pf-color-black-600 ← Palette color

↳ #737679 ← Hex value

PatternFly content enhancements: Style guide

1. Medium
2. Twitter
3. GitHub documentation
4. Home page copy
5. Style guide





Design guidelines

Content

Style guide
Voice
Writing

3 parts of the current style guide:

Style

- Mechanics
- Capitalization, punctuation, and more

Voice

- Being clear, concise, useful, conversational, consistent
- Writing for all audiences

Writing

- UX writing in the design process
- UX writing tips

Identify need

- In-depth UX writing details
- Template for Flyers
- Single source of truth

Create solution

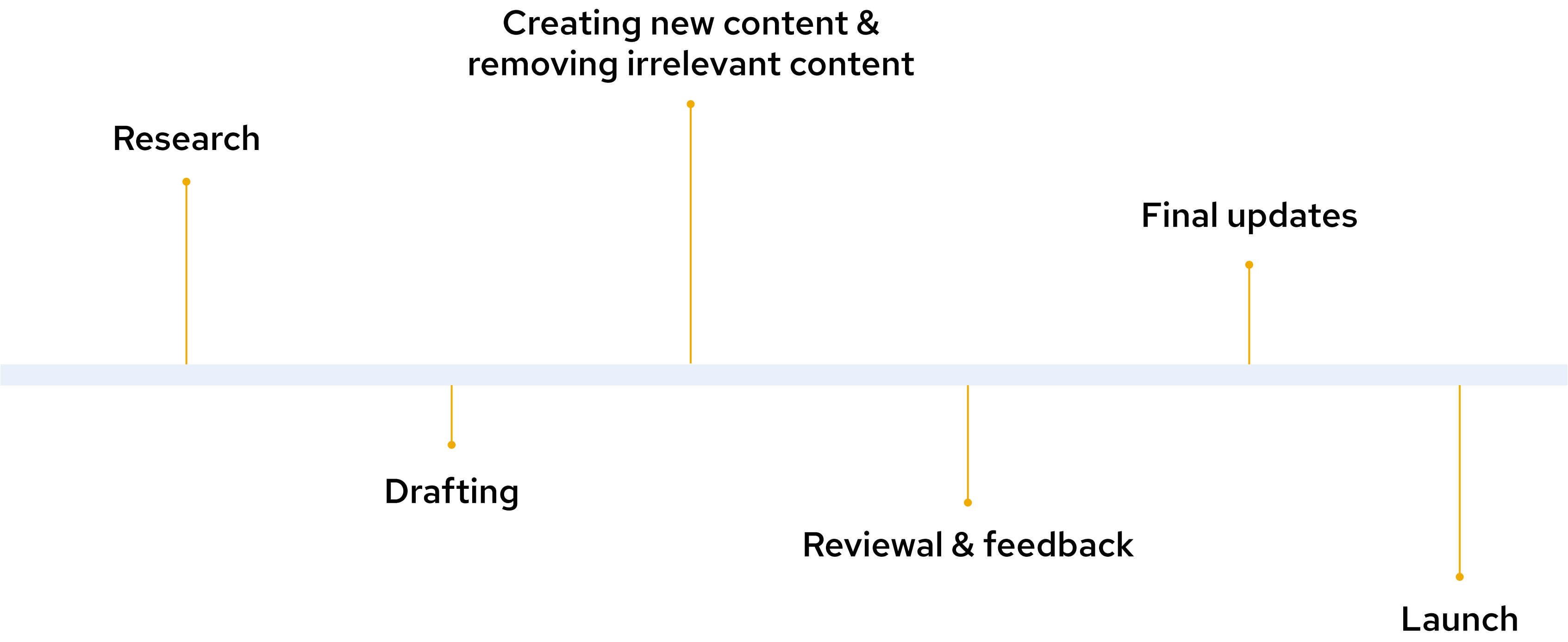
- Feedback
- Research
- Group decision

PatternFly style guide

Unify all UX writing guidelines into a single PatternFly style guide.

Expand upon what we have to include more in-depth, comprehensive UX writing guidelines and best practices.

Strengthen our content community.



Research

Drafting

Creating new content & removing irrelevant content

Reviewal & feedback

Final updates

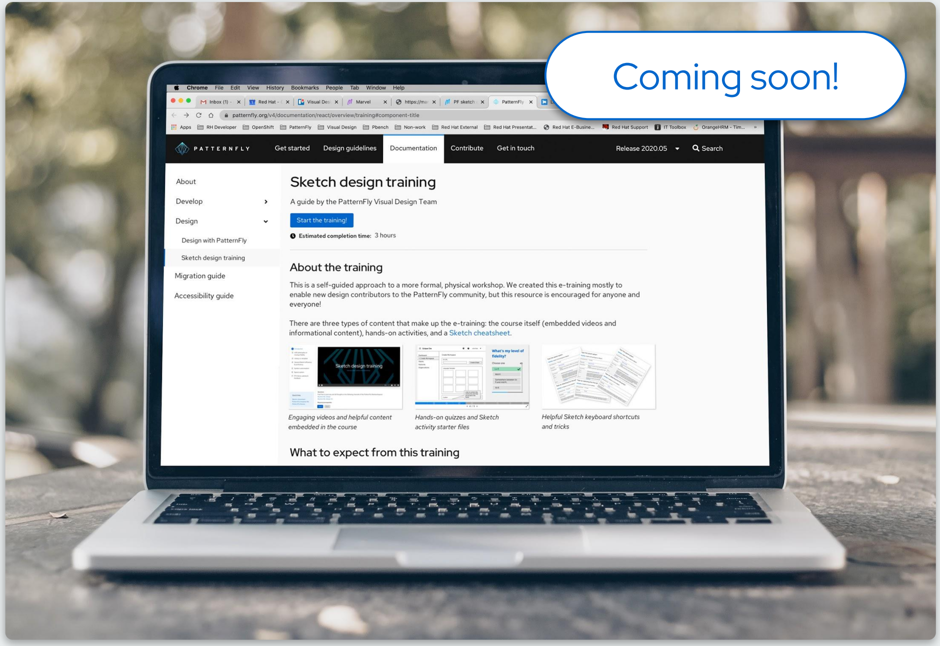
Launch

Thank you!

Sketch design
e-training for PatternFly

Sketch design e-training for PatternFly

A resource by the PF visual design team



Coming soon!

Sketch design training

A guide by the PatternFly Visual Design Team

Estimated completion time: 3 hours

About the training

This is a self-guided approach to a more formal, physical workshop. We created this e-training mostly to enable new design contributors to the PatternFly community, but this resource is encouraged for anyone and everyone!

There are three types of content that make up the e-training: the course itself (embedded videos and informational content), hands-on activities, and a Sketch cheatsheet.

Engaging videos and helpful content embedded in the course

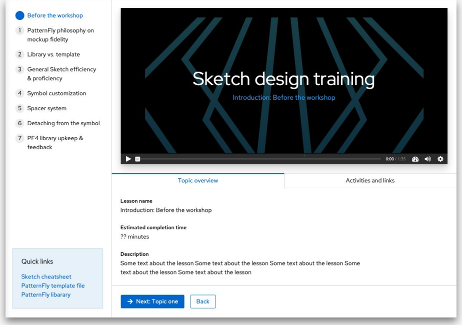
Hands-on quizzes and Sketch activity starter files

Helpful Sketch keyboard shortcuts and tricks

What to expect from this training

A self-guided approach to in-person workshops with...

Helpful videos and content embedded in the course



- Before the workshop
- 1 PatternFly philosophy on mockup fidelity
- 2 Library vs. template
- 3 General Sketch efficiency & proficiency
- 4 Symbol customization
- 5 Spacer system
- 6 Detaching from the symbol
- 7 PF4 library upkeep & feedback

Sketch design training

Introduction: Before the workshop

0:00 / 1:35

Topic overview | Activities and links

Lesson name
Introduction: Before the workshop

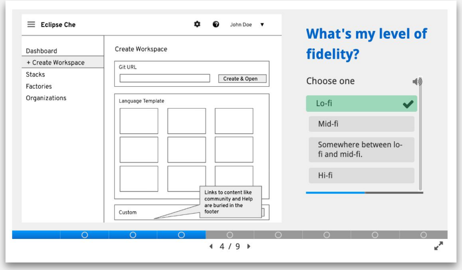
Estimated completion time
?? minutes

Description
Some text about the lesson Some text about the lesson Some text about the lesson Some text about the lesson Some text about the lesson

Quick links

Sketch cheatsheet
PatternFly template file
PatternFly libarary

Hands-on quizzes and Sketch activity starter files



Links to content like community and Help are buried in the footer

What's my level of fidelity?

Choose one

- Lo-fi ✓
- Mid-fi
- Somewhere between lo-fi and mid-fi.
- Hi-fi

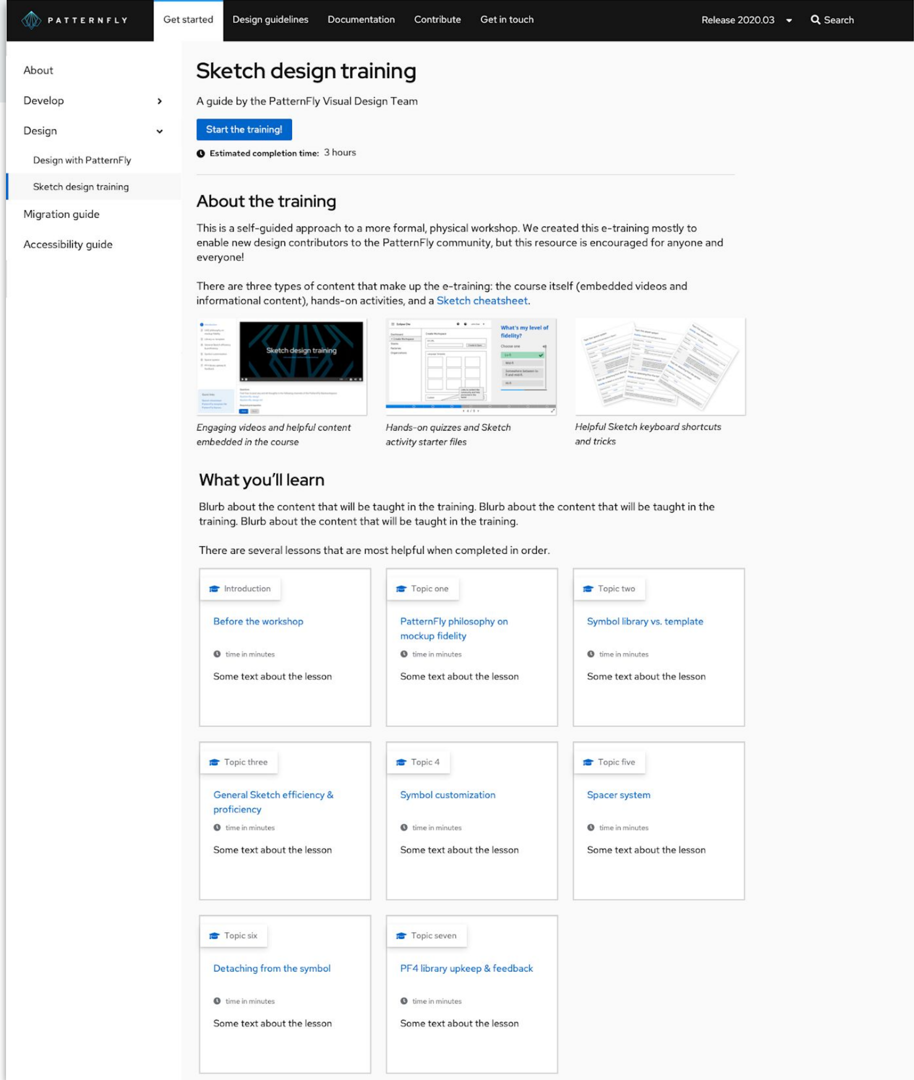
4 / 9

Helpful Sketch keyboard shortcuts and tricks

Topic five: spacer system
Activity: create a full card layout in Sketch
Topic six: detaching from the symbol
Activity: to detach or not to detach

Current direction for what’s to come…

Check out a sneak peak



Sketch design training

A guide by the PatternFly Visual Design Team

Estimated completion time: 3 hours

About the training

This is a self-guided approach to a more formal, physical workshop. We created this e-training mostly to enable new design contributors to the PatternFly community, but this resource is encouraged for anyone and everyone!

There are three types of content that make up the e-training: the course itself (embedded videos and informational content), hands-on activities, and a Sketch cheatsheet.

Engaging videos and helpful content embedded in the course

Hands-on quizzes and Sketch activity starter files

Helpful Sketch keyboard shortcuts and tricks

What you’ll learn

Blurb about the content that will be taught in the training. Blurb about the content that will be taught in the training. Blurb about the content that will be taught in the training.

There are several lessons that are most helpful when completed in order.

Introduction
Before the workshop
time in minutes
Some text about the lesson

Topic one
PatternFly philosophy on mockup fidelity
time in minutes
Some text about the lesson

Topic two
Symbol library vs. template
time in minutes
Some text about the lesson

Topic three
General Sketch efficiency & proficiency
time in minutes
Some text about the lesson

Topic 4
Symbol customization
time in minutes
Some text about the lesson

Topic five
Spacer system
time in minutes
Some text about the lesson

Topic six
Detaching from the symbol
time in minutes
Some text about the lesson

Topic seven
PF4 library upkeep & feedback
time in minutes
Some text about the lesson

Any questions?

PatternFly Website Usability Results

Background

- This study was conducted as the second part of an overall effort to evaluate the usability of the PatternFly website.
    - First portion looked at the masthead and sidebar navigation structures.
- The PatternFly team was primarily interested in the effectiveness of the documentation presented on the PF website.
    - Also interested in mobile use and usability of search functionality.

Method

- This study used a semi-structured interview method to obtain qualitative data about users' experience with the PF website
- 6 participants were recruited internally.
- Interviews lasted approximately 30-45 minutes each
- 4 participants classified themselves as designers
  - Other 2 classified as developers.
- Questions ranged from basic information about participant's role, use of the website in mobile devices, use of the search function, and use of the documentation.
- At the end, participants were presented with an alternative mockup of the documentation and asked their thoughts.

Result Highlights

Documentation - Does it meet your needs?

- 2 participants (1 designer, 1 developer) said the documentation meets their needs
  - 4 participants said it does not.
- Primary reason for this was related to the examples
  - 3 participants (2 designers, 1 developer) mentioned that the examples are isolated and lack context about how the components fit together with each other on a page.
    - “I wish there were more demos, examples in context. Wish more things were clickable. More real life actual code examples to see the interactions.”
  - 2 participants (1 designer, 1 developer) mentioned that there is a significant gap between the design guidelines and documentation
    - Some components exist in the documentation and not the design guidelines
      - “The divorced nature of the design guidelines from the widgets makes it harder for me to use. Certain widgets exist only as HTML or React but not both”
      - “There’s a gap between the design guidelines and documentation with no link between them”

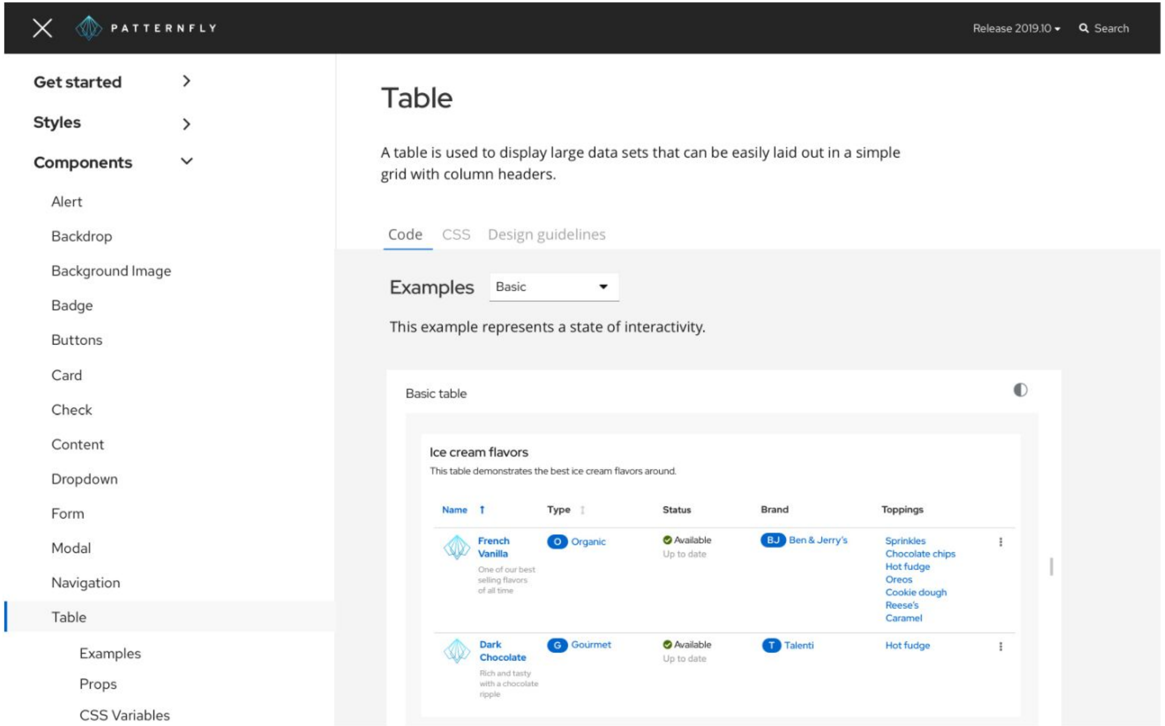
Documentation - Pain Points

- Two major pain points observed
  - Not enough examples shown in context.
    - Participants suggested multiple solutions...
      - Having multiple versions of examples, one isolated, and one in context to see interactions.
      - Having a dummy app with all components built to see how each one works.
  - Having to navigate back and forth between design and usage of components.
    - Participants were frustrated with having to go back and forth between design guidelines and documentation for the same component.
- Designers do not necessarily know the difference between HTML and React
  - Need more explanation around each one.

Alternative Mockup

- Participants were shown an alternative mockup at the end of the interview and asked their thoughts.
- The alternative tries to address some of the pain points that people had mentioned
    - Also based off of the results from the first phase of the study on the navigation structure.

Alternative Mockup



Table

A table is used to display large data sets that can be easily laid out in a simple grid with column headers.

Examples

This example represents a state of interactivity.

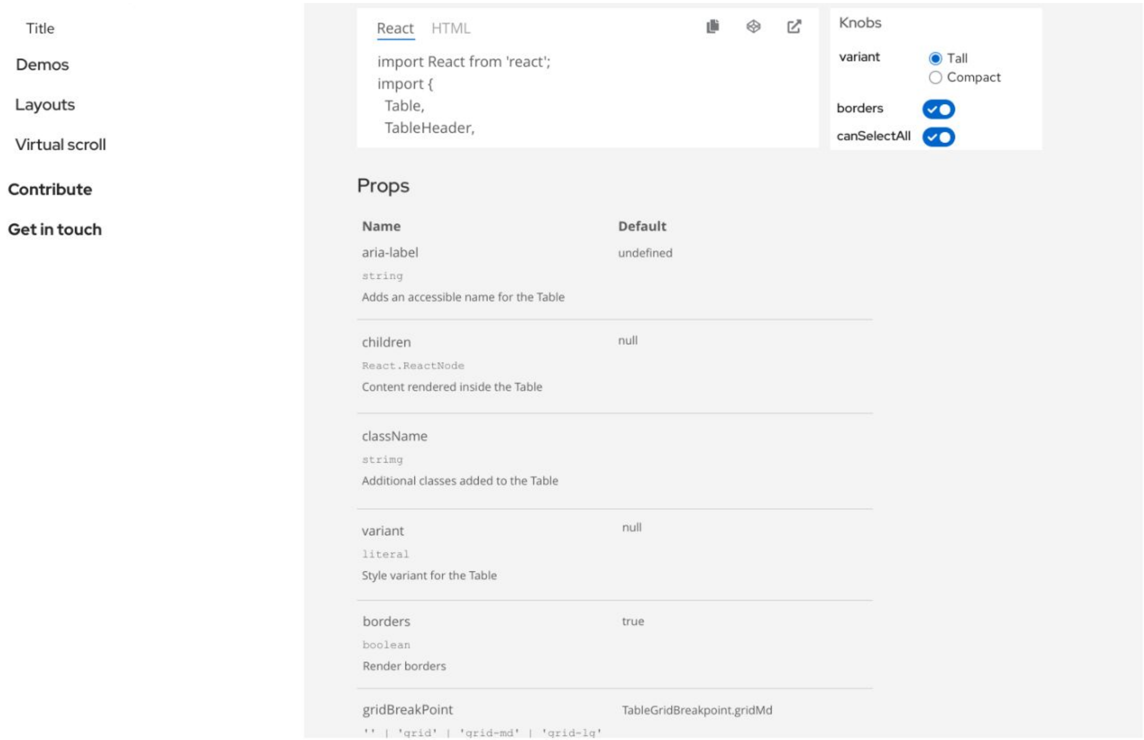
Basic table

Ice cream flavors

This table demonstrates the best ice cream flavors around.

Name ↑	Type	Status	Brand	Toppings
French Vanilla One of our best selling flavors of all time	O Organic	✓ Available Up to date	BJ Ben & Jerry's	Sprinkles Chocolate chips Hot fudge Oreos Cookie dough Reese's Caramel
Dark Chocolate Rich and tasty with a chocolate ripple	G Gourmet	✓ Available Up to date	T Talenti	Hot fudge

Alternative Mockup



Title

Demos

Layouts

Virtual scroll

Contribute

Get in touch

```
import React from 'react';
import {
  Table,
  TableHeader,
```

Knobs

variant ● Tall ○ Compact

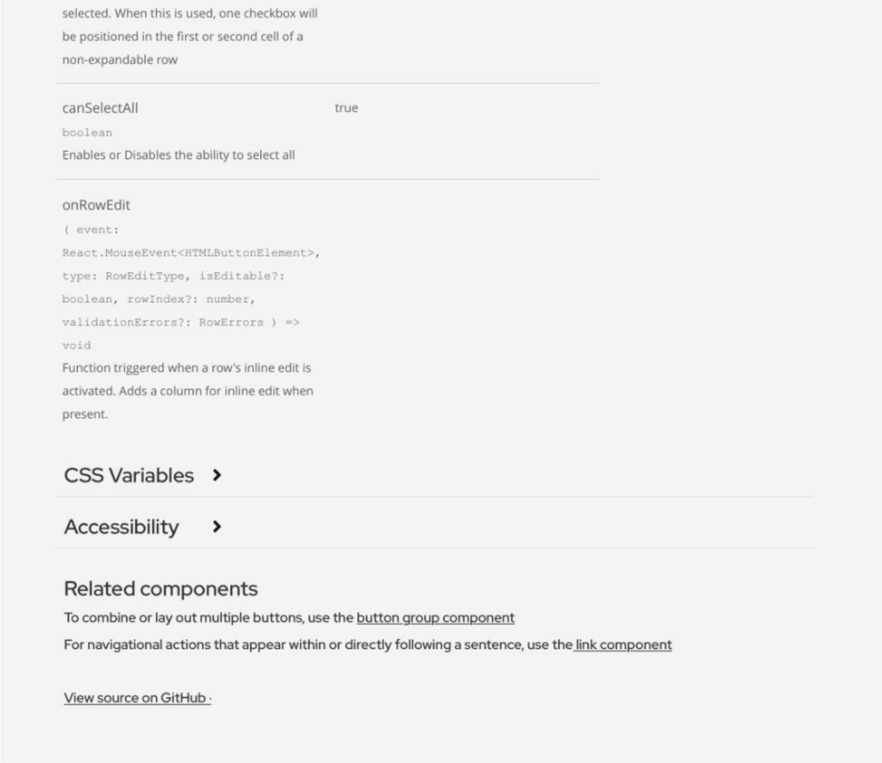
borders ✓

canSelectAll ✓

Props

Name	Default
aria-label `string` Adds an accessible name for the Table	undefined
children `React.ReactNode` Content rendered inside the Table	null
className `strimg` Additional classes added to the Table	
variant `literal` Style variant for the Table	null
borders `boolean` Render borders	true
gridBreakPoint `'' | 'grid' | 'grid-md' | 'grid-lg'`	TableGridBreakpoint.gridMd

Alternative Mockup



selected. When this is used, one checkbox will be positioned in the first or second cell of a non-expandable row

canSelectAll true

`boolean`

Enables or Disables the ability to select all

onRowEdit

```
( event:
React.MouseEvent<HTMLButtonElement>,
type: RowEditType, isEditable?:
boolean, rowIndex?: number,
validationErrors?: RowErrors ) =>
void
```

Function triggered when a row's inline edit is activated. Adds a column for inline edit when present.

CSS Variables ›

Accessibility ›

Related components

To combine or lay out multiple buttons, use the button group component

For navigational actions that appear within or directly following a sentence, use the link component

View source on GitHub

Alternative Mockup

- Overall, participants felt that this was an improvement over the current design.
  - “Yes, this makes a lot more sense, having it broken down into design and develop. I kinda like it.”
  - “I think this would work better for me. It feels like a more simplified navigation.”
- There were some issues that participants mentioned though
  - “Hard to tell what is UI itself and what is examples.”
  - “Pull components up to a top level nav category.”
  - “Not sure about the button types dropdown. Might be too granular.”
  - “Examples might be trickier to navigate. Is there a way to see all examples at once?”
  - “I would probably push carats all the way to the right.”
  - “May want to consider changing the "Code" section title to something else.”
  - “New Props section seems harder to read than old one”
  - “I would like an indicator that says which props are required or not”

Any questions?

Thank you!

What we’re working on next

- Working on major release
- Helpful link to follow our release schedule - Zack Allen

Q&A