<table>
<thead>
<tr>
<th>CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>GETTING STARTED</td>
</tr>
<tr>
<td>CONTENT FORMATTING</td>
</tr>
<tr>
<td>LABELS</td>
</tr>
<tr>
<td>INPUTS</td>
</tr>
<tr>
<td>BUTTONS</td>
</tr>
<tr>
<td>LAYOUTS</td>
</tr>
<tr>
<td>DESIGN PATTERNS</td>
</tr>
<tr>
<td>MOBILE DESIGN</td>
</tr>
<tr>
<td>CHARTS</td>
</tr>
<tr>
<td>BRANDING</td>
</tr>
<tr>
<td>NEWS EVENT ICONS</td>
</tr>
<tr>
<td>INDICATOR ICONS</td>
</tr>
<tr>
<td>RECORD ICONS</td>
</tr>
</tbody>
</table>
The Appian Style Guide for Interface Designers describes recommended best practices for creating attractive and usable interfaces. Study the topics in this document to understand how to make the best choices and avoid common mistakes when designing for a great user experience.

Great user interfaces aren't just about functionality.

Is the appearance contemporary and harmonious?

Can new users quickly learn how to do their work?

Can activities be completed with a minimum of steps and time?
Consciously design each interface to best meet the needs of its intended audience.

Provide enough information to allow new users to quickly complete tasks without additional help. At the same time, remove unnecessary content that may reduce efficiency when performing repeated activities.

“What would users need that’s not here?”

“What’s here that users don’t really need?”
Minimize the number of clicks and keypresses

Minimize scrolling and mouse movement by positioning controls close to their associated content

Minimize the amount of content comprehension required to complete a task

Optimize sequences that would hamper productivity for a frequent user

GETTING STARTED
Create New Purchase Request

Order Details
Requested By

SUBMIT REQUEST

✓ DO

Use title case capitalization for titles, labels, and headers. For example:
• Page/site titles
• Section headings
• Prompt/dialog titles
• Field labels
Use sentence case capitalization for messages, instructions, and descriptions. For example:

- Form and field instructions
- Prompt dialog messages
- Record, report, and action descriptions
When an instruction or description is composed of a single sentence, do not include a period at the end of the sentence.

When there is more than one sentence, all sentences should have a period.
Use titles with data specific to the form instance rather than generic form titles. The action form title should match the title that appears in the actions list exactly.
Use concise and self-explanatory titles. If not possible, add a description and make sure it is not redundant.

Avoid restating the object type (i.e. record, report, action) in the description.
Avoid displaying information intended to assist users with data input (i.e. required indicator, instructions) on read-only fields.

Use read-only text instead of a disabled dropdown component to display a selected dropdown value on a dashboard.
**NUMBER FORMAT**

- Use digit group separators for long numbers to increase readability. This does not apply to ID numbers.
- Show the appropriate level of precision necessary given the purpose of the data.
- Use consistent number precision (even if “.00”).

<table>
<thead>
<tr>
<th>Content</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Seconds Active</strong></td>
<td>1,670.6</td>
</tr>
<tr>
<td><strong>Total Cost</strong></td>
<td>$236.90</td>
</tr>
<tr>
<td><strong>Average Cost</strong></td>
<td>$200.00</td>
</tr>
</tbody>
</table>

**DO**

- Correctly formatted numbers with digit group separators.

**DON'T**

- Incorrectly formatted numbers without digit group separators.

- Incorrectly formatted numbers with unnecessary zeros after the decimal point.
When displaying a date/time in a read-only text field, show times in the user’s configured timezone to eliminate the need to display the timezone abbreviation or offset.

**DO**

- Time Activated: 8/20/2015 3:20 PM

**DON'T**

- Time Activated: 8/20/2015 3:20 PM GMT +00:00

The timezone for the above date and time is GMT.
<table>
<thead>
<tr>
<th>Item</th>
<th>Department</th>
<th>Date</th>
<th>Qty</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop</td>
<td>Technology</td>
<td>12/03/2016</td>
<td>10</td>
<td>$5,000</td>
</tr>
<tr>
<td>Pen</td>
<td>Sales</td>
<td>1/1/2017</td>
<td>50</td>
<td>$400</td>
</tr>
<tr>
<td>Sticky Notes</td>
<td>Sales</td>
<td>1/17/2017</td>
<td>100</td>
<td>$300</td>
</tr>
</tbody>
</table>

**GRID ALIGNMENT**

Left-align text and right-align numbers and dates (in left-to-right locales).

Align headers consistently with column content.
Field labels may be shown above the component, adjacent to the component, or not shown at all.

Note that labels are always shown above the component in the mobile application.
The above-component label position generally works well for form input fields (like textboxes and dropdowns).

For all components, labels above are especially preferable to adjacent when:
- The component is wide (e.g. grids & charts), since adjacent labels take up horizontal space
- The label text is long (to avoid wrapping)
Show labels adjacent to the component when:

- Displaying non-editable values, such as record attributes (to enhance visual grouping of labels vs. values, especially when some of the values may be blank)
- The interface has many fields (to minimize vertical scrolling)
Avoid mixing different label positions within the same interface or section as this creates an unbalanced layout.

Consider the guidelines for determining label positioning and choose the option that best balances the requirements of all fields.
Field labels may be excluded if they would be redundant. For example, if:

- An interface contains a single grid or chart and the page title sufficiently describes it
- A section header label sufficiently describes a group of related fields

Be careful of the accessibility impact of excluding labels. Assistive technologies may expect a text label description of each field.
Avoid repeating words when labeling a group of related inputs.

Use a section header to provide context, allowing for more concise field labels.

<table>
<thead>
<tr>
<th>Award Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Title</strong></td>
</tr>
<tr>
<td><strong>Recipient</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Award Title</th>
<th>Award Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Award Recipient</strong></td>
<td></td>
</tr>
</tbody>
</table>

**REDUNDANT LABELS**

- ✓ **DO**
- ✗ **DON'T**
Avoid using a colon ("::") after a field or section label.

Use consistent capitalization in labels. Title case is recommended.
Avoid using labels with a conversational tone if other labels on the form are concise and direct.
When displaying rich text header, use the above label position or exclude the label for proper alignment.

Field labels may be excluded if the headers are sufficient to describe or organize the content.
Use descriptive display text for link labels.

Avoid unnecessary or redundant words like “link.”

A URL should not be displayed as the link label unless there is an explicit reason for users to see the URL.

DON'T

- Link to Timesheets
- Link to Benefits
- Link to Calendar

Supporting Document
https://docs.google.com/presentation/d/11QAVCf9IOJMNQxO9U3O1FpgqAIHoKycrHOhAYPgWC0/edit#slide=id.g54233b027_0106
INPUTS
Use a radio button group to provide users with a short (<5 items) list of choices from which they must select one.

Don't show a radio button group with no default-selected button.

The compact layout option should only be used for radio groups containing short choice labels that are not likely to wrap onto the next line.
Use a checkbox group to provide users with a short (<5 items) list of choices from which none, one, or many items may be selected.

Use a single checkbox to provide users with a Boolean toggle.
Use a dropdown to provide users with moderately-long lists of choices from which one or many may be selected, depending on configuration.

Use radio/checkbox groups for shorter lists as all choices can easily be seen by the user.

Use pickers when browsing is impractical.
Select the appropriate paragraph height ("short", "medium", or "tall") based on the expected length of user responses.

Use the "short" paragraph height in editable grids to align paragraphs with dropdown, number, and text inputs in adjacent columns.
INSTRUCTION

You should use instructions to display important content that is essential to completing the task.

If the content should always be visible, use instructions as opposed to a help tooltip or placeholder.

Consider using instructions when a field label becomes too lengthy.
Consider using a help tooltip instead of instructions for content that does not need to be read each time a user views the form.

For example, a help tooltip is appropriate for showing instructions that are most useful to new users.
DO

Account Number
The 6 digit number on your invoice

DON'T

Account Number
135792

PLACEHOLDER TEXT

Use placeholder text to describe the correct input format or provide a brief hint to assist with value input.

Placeholder text should not replace field labels.
Use placeholder text for picker components to provide a distinction from regular text inputs.

In general, use sentence case capitalization and keep the message as short as possible.
BUTTONS
Primary button styling draws attention to the most common action on an interface to speed up user interactions. Primary and normal-styled buttons use the configured accent color.

Destructive button styling highlights actions that result in loss of persisted data. By default, use normal button styling.

Don't rely on the styling of buttons to convey their meaning. Use text labels that convey sufficient information to users who cannot see the button color.
Don’t show more than one primary button on an interface.

Don’t use primary styling for buttons that delete data or cancel the user’s current activity.

Assume that many users will be biased toward selecting the primary button; make sure to limit side effects of mistakes.
Don’t use the destructive style for easily-reversible actions or the removal of information entered by the user while viewing the interface.

Don’t show both primary and destructive buttons in one interface. Include a maximum of either one primary button or one destructive button.
If possible, use a verb that better represents the button action (e.g. “Approve”) instead of a generic label (e.g. "Submit").

For wizards, use a “Next” or “Continue” label to indicate that additional steps remain.
The form footer button group is only for buttons that submit an entire form or navigate away from the form (Cancel, Go Back, etc.).

Use inline button groups within the interface content for buttons that act on part of the content and do not take the user away from the interface (e.g. buttons as a toolbar for selected items in a grid).
Place all form submission buttons on the right side of the button group. The most commonly-used button should come first (left-most). This button should use the primary style (unless the action deletes persisted data, in which case it should use the destructive style).

Go back/cancel buttons should be placed on the left side of the button group (back button left-most).
Buttons that are temporarily unavailable due to the state of form data should generally be disabled, not hidden.

However, if the availability of a large number of buttons changes as users interact with the form, unavailable buttons should be hidden to reduce clutter and allow users to easily see valid options.
Use concise titles for related actions to prevent shortcut button label truncation. If additional text is needed to convey the purpose of the action, add descriptive text rather than lengthening the title.

Make only the most relevant related actions to a record view available as shortcuts, no more than 3 if possible.
LAYOUTS
Use a one-column layout only when the content is wide and will look best when spanning the full width.

Don't use a one-column layout when the content does not require much horizontal space. Consider using a two-column layout and leaving the second column empty (if there isn't enough content to fill it) to avoid showing excessively wide fields.
Use a two-column layout for maximum density when showing content that does not require a lot of horizontal space. Be careful of the effect of different label positions on available width.

When arranging related items across two columns, note that it is most natural for users to scan from top to bottom within one column before moving on to the next.

**Active Employees**

<table>
<thead>
<tr>
<th>Name</th>
<th>Title</th>
<th>Department</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Smith</td>
<td>Director</td>
<td>Engineering</td>
<td>5/20/22</td>
</tr>
<tr>
<td>Michael</td>
<td>Analyst</td>
<td>Finance</td>
<td>6/3/22</td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td>Software</td>
<td>Engineering</td>
<td>7/4/22</td>
</tr>
<tr>
<td>Reed</td>
<td>Engineer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Angela</td>
<td>Sales</td>
<td>Sales</td>
<td>7/27/22</td>
</tr>
<tr>
<td>Cooper</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elizabeth</td>
<td>Sales</td>
<td>Sales</td>
<td>12/12/22</td>
</tr>
<tr>
<td>Ward</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Inactive Employees**

<table>
<thead>
<tr>
<th>Name</th>
<th>Title</th>
<th>Department</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Smith</td>
<td>Director</td>
<td>Engineering</td>
<td>5/20/22</td>
</tr>
<tr>
<td>Michael</td>
<td>Analyst</td>
<td>Finance</td>
<td>6/3/22</td>
</tr>
<tr>
<td>Johnson</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td>Software</td>
<td>Engineering</td>
<td>7/4/22</td>
</tr>
<tr>
<td>Reed</td>
<td>Engineer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Angela</td>
<td>Sales</td>
<td>Sales</td>
<td>7/27/22</td>
</tr>
<tr>
<td>Cooper</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elizabeth</td>
<td>Sales</td>
<td>Sales</td>
<td>12/12/22</td>
</tr>
<tr>
<td>Ward</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Minimize white space by keeping columns well-balanced and using an appropriate label position.
DESIGN PATTERNS
**WIZARDS**

Use wizards to guide users through sequential steps for completing a task. Wizards also help to break long forms up into more manageable components.

Use a milestone component to show the user’s progress.

Don’t design wizards that require users to frequently reference previous steps to complete the current step.

<table>
<thead>
<tr>
<th>Order</th>
<th>Shipping</th>
<th>Payment</th>
<th>Confirm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Street Address</strong></td>
<td><strong>City</strong></td>
<td><strong>State</strong></td>
<td><strong>Zip Code</strong></td>
</tr>
<tr>
<td><strong>Apt/Unit</strong></td>
<td></td>
<td><strong>-- Select a State --</strong></td>
<td></td>
</tr>
</tbody>
</table>

[BACK TO ORDER] [CANCEL] 

[CONTINUE TO PAYMENT]
Use a single form, instead of a series of smaller forms, if data can be entered non-sequentially or if users must view content concurrently.

Use sections to group related information, especially on longer forms.

Keep in mind that including a large number of interface elements on a single form may impact performance.
Provide actions on grid items by showing a left-aligned button group directly above the grid.

Show only the buttons representing actions available for the currently-selected grid rows. (Special case: If it’s possible to have a state where no buttons are available, disable buttons instead of hiding them.)
Consider using inline actions on each grid row if:

- There are not enough actions to justify having checkboxes and a toolbar
- Users will typically be taking action on one row item at a time

<table>
<thead>
<tr>
<th>Question</th>
<th>Type</th>
<th>Req’d</th>
<th>Advanced Setup</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>What is your age?</td>
<td>Single selection from list</td>
<td>▼</td>
<td>☐</td>
<td>(\textbf{Choices:} \ 1; 2; 3; 4; 5)</td>
</tr>
<tr>
<td>What is your gender?</td>
<td>Single selection from list</td>
<td>▼</td>
<td>☐</td>
<td>(\textbf{Choices:} \ 1; 2; 3; 4; 5)</td>
</tr>
<tr>
<td>Where do you live?</td>
<td>Text</td>
<td>▼</td>
<td>☐</td>
<td></td>
</tr>
</tbody>
</table>
Buttons are for taking action, links are for navigation.

However, form footer buttons are appropriate for submission, cancellation, and stepping back even though there is a navigation result to those actions.

Links may be used for localized actions such as expanding collapsed text (where a button would be too prominent).

"Because it ain't in the books so—that's why. Now, Ben Rogers, do you want to do things regular, or don't you?—that's the idea. Don't you reckon that the people that made the books know what's the correct thing to do? Do you reckon you can learn 'em anything? Not by a good deal. No, sir, we'll just go on and ransom them in the regular way."

"All right. I don't mind; but I say it's a fool way, anyhow. Say, do we kill the women, too?"

"Well, Ben Rogers, if I was as ignorant as you I wouldn't let on. Kill the women? No; nobody ever saw anything in the books like that. You fetch them to the cave, and you're always as polite as pie to them; and by and by they fall in love with you, and never want to go home any more."

"Well, if that's the way I'm agreed, but I don't take no stock in it. Mighty soon we'll have the cave so cluttered up with women, and fellows waiting to be ransomed, that there won't be no place for the robbers. But go ahead, I ain't got nothing to say."

Little Tommy B...More
Elements that conditionally show/hide based on user input should be visually grouped and displayed directly below the associated toggle.

In the “DO” example above, shown/hidden elements are placed in a section to distinguish them from other unrelated elements on the form.
Use concise titles for multi-page sites. Keep in mind that there is even less horizontal space on a mobile device.
MOBILE DESIGN
Multi-column layouts are flattened into a single column on phones and small tablets.

Make sure that your design doesn't only make sense when certain fields are side-by-side.
When using the iOS or Android mobile app, button layouts are flattened into a single column on phones and small tablets, with primary buttons appearing above secondary buttons.

Make sure that the button order makes sense in this alternate layout.
While concise labels and instructions are always recommended, it’s particularly important to reduce clutter, wrapping, and scrolling on mobile screens.

Certain components, by definition, may be configured to require a lot of screen real estate (e.g. milestones with many steps, grids with many columns). Avoid these situations if you’re targeting narrow screens.
Avoid design patterns that assume a very specific rendering behavior, like images laid out in a particular way to simulate tabs.

Each mobile platform renders components differently and SAIL only guarantees logical consistency, not an exact replica on every client.
Pie charts should only be used to represent data that can be placed into distinct categories and should not consist of more than 5 or 6 slices.

Always sort the data values in ascending or descending order and include value and category labels for the slices.
Use bar charts for direct comparison of data or to show data over time when the number of time intervals is small.

Use a bar chart instead of multiple pie charts when comparing two sets of data.

Sort columns in systematic order, either by size of value or by sequential categories (e.g. sorting by year).
Line charts are best used for presenting data over time and are more effective than bar charts for presenting many data points.

Conventionally, the x-axis contains the categories of time, and the y-axis the frequencies of the measured data.

Charts with more than 5 lines tend to be confusing unless the lines are well separated.
BRANDING
A configurable accent color is used to highlight key UI elements in Tempo such as primary-style buttons, links, and section headings.

Avoid accent colors that are:

- Too close to the black normal text color
- Too close to the red destructive button and error message color
- Too low in contrast against the white page background. The accent color should have a minimum contrast ratio of 4.5:1. Use a contrast checker to ensure your selected color meets the requirement.
BRANDING

Do

SITE COLORS

Use a site logo with a transparent background and sufficient contrast against the navigation bar color.

A dark gray or white text and icon color is automatically applied based on the selected background colors. Avoid medium-darkness background colors which may not provide sufficient contrast.
The a!iconNewsEvent() system function provides a convenient way for designers to access a library of typical news event icons.

Use of these standard icons is highly encouraged as they were designed to promote a consistent and harmonious interface experience.
The recommended dimensions for custom news event icons are 134px (h) x 124px (w). This size allows for sharp rendering of icons on high-pixel-density screens such as Apple’s Retina displays.

The border thickness for the standard icon template is 6px. The PNG image format is recommended.
Use the recommended color palette to maximize interface consistency. Each icon should use a single, solid color (no transparency) on a white background.

When using icon color to indicate business meaning (e.g. red-problem) make sure to provide the same information in event text for the benefit of color-blind and non-sighted users. Never use different colored icons for the same event for decorative purposes.
The a!iconIndicator() system function provides a convenient way for designers to access a library of typical indicator icons.

“Icon” size should be specified for indicator icons.
The recommended dimensions for custom indicator icons are 40px (h) x 40px (w).

For proper visual balance, an icon shape of approximately 32 x 32px is recommended, centered within a transparent background.

Use solid colors and simple, recognizable shapes.
Use indicator icons as a concise representation of the state of items shown on a grid. The domain of possible icons should be small and well-known to users.

Make sure to provide an alternate text description of each icon for non-sighted users or users who do not recognize what an icon represents.

<table>
<thead>
<tr>
<th>Task</th>
<th>Date</th>
<th>Assigned To</th>
<th>Icon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implement SSI Setup</td>
<td>1/01/2016</td>
<td>Robert Skye</td>
<td>✓</td>
</tr>
<tr>
<td>Onboard New Hires</td>
<td>5/20/2016</td>
<td>Robert Skye</td>
<td>❗</td>
</tr>
<tr>
<td>Plan Team Meeting</td>
<td>7/11/2016</td>
<td>Robert Skye</td>
<td>❌ Overdue</td>
</tr>
</tbody>
</table>
The recommended dimensions for custom record icons are 124px (h) x 124px (w). This size allows for sharp rendering of icons on high-pixel-density screens such as Apple’s Retina displays.

The rendered, visible bounds of the displayed icon are 62(h) x 62(w) device-independent pixels (dip).
Record icons should feature simple shapes that are easily recognized at small display sizes. Use a transparent or white background for empty areas of the icon and avoid adding borders.

Photorealistic or highly-detailed images are not appropriate for icons.
Use an icon that represents a distinguishing attribute of the record, such as a company logo. This will allow users to quickly identify a record by viewing the associated icon.

Avoid displaying the same icon for every record, as this does not help users scan through the list.

**DON'T**

- *Printer and Scanner*
  Requestor: Jane Doe | Washington, DC

- *Conference Phone in Main Meeting Room*
  Requestor: John Smith | New York

- *Front Desk Office Supplies*
  Requestor: Michael Johnson | Washington, DC

- *Office Supplies for Marketing team*
  Requestor: Rebecca Smith | Washington, DC
Version 6, January 2017

➢ Added the following sections
  ○ Branding
➢ Added the following topics
  ○ Accent Color
  ○ Site Colors
➢ Updated screenshots to show 17.1 component styling
➢ Removed Wide Icons guidance from Record Icons section (not applicable in Appian 17.1)
Version 5, April 2016

➢ Reordered sections (most notably, moved icon-related sections to the end)
➢ Added the following sections
  ○ Mobile Design
➢ Added the following topics
  ○ Period Usage
  ○ Action Form Titles
  ○ List View Items
  ○ Read Only Format
  ○ Label Format
  ○ Consistent Tone
  ○ Picker Placeholder Text
  ○ Sites Tabs
  ○ Flattened Columns
  ○ Flattened Buttons
  ○ Wrapping & Scrolling
  ○ Logical Consistency
➢ Renamed Format section to Content Formatting
➢ Updated Sentence Case topic to include record, report, and action descriptions as an example of when to use sentence case capitalization
➢ Changed the “don’t” example to a “do” for Title Case and Sentence Case topics
➢ Added a second “don’t” example to the Date/Time Format topic
➢ Updated Button Styles topic to advise against relying on the styling of buttons to convey their meaning
➢ Updated images for Button Availability, Sections, and Inline Grid Actions topics
Version 4, October 2015

- Updated to a 16:9 widescreen layout from the 4:3 layout
- Added the following topics
  - Paragraph Fields
  - Placeholder Text (new in Appian 7.11)
  - Help Tooltip (new in 7.11)
  - Date/Time Format
  - Inline Grid Actions
- Updated Icon Library topics (both Indicator and News Event) to include the system rules used to call the icons (new in 7.9)
- Condensed Icon Styling topic into one page
- Updated Instruction Text topic to provide guidance on how to choose between using instructions, placeholder text, or a help tooltip
- Updated image on Wizards topic to use blue milestone bar (updated in 7.10), added a link in the milestone component (new in 7.11), and gave buttons more descriptive labels
- Updated Grid Toolbars topic to differentiate it's usage from Inline Grid Actions
- Added a “don’t” example for Progressive Disclosure topic
- Updated image for Button Labels, Button Location, Button Availability, and Instruction Text topics
Version 3, April 2015
➢ Added the following sections
  ○ Getting Started
  ○ Format
➢ Added the following topics
  ○ Using This Guide
  ○ Intuitive & Efficient Design
  ○ Title Case
  ○ Sentence Case
  ○ Number Format
  ○ Grid Alignment
➢ Updated Radio Buttons topic to include compact layout option (new in Appian 7.9)
➢ Updated Icon Library topics to show icon selection for designers (new in Appian 7.9)
➢ Renamed System Event Icons topic to News Event Icons
➢ Renamed Grid Icons topic to Indicator Icons
Version 2, December 2014
➢ Added the following topics
  ○ Icon Usage
  ○ Redundant Labels
  ○ Rich Text Headers
  ○ Link Labels
  ○ Button Labels
  ○ Related Action Shortcuts
  ○ Instructional Text
  ○ White Space
  ○ Sections
  ○ Progressive Disclosure
  ○ Charts (Pie, Bar, Line)
➢ Renamed Field Labels section to Labels
➢ Renamed Patterns section to Design Patterns
➢ Added Contents page