

Vault Safety Writing Guide

This guide provides information to help you write Vault Safety online help documentation.

- [Introduction to the Authoring Environment](#)
- [How to Add a New Page](#)
 - [Naming Files and Folders](#)
 - [Add YAML Front Matter](#)
 - [Add a Table of Contents](#)
 - [Add Page Content](#)
 - [Update the Navigation Menu](#)
 - [Add a Top-Level Link](#)
 - [Add a Sublink](#)
 - [Update the Pagination List \(Previous and Next Buttons\)](#)
- [Writing Style](#)
 - [Titles and Headings](#)
 - [Punctuation](#)
 - [Commas](#)
 - [Colon](#)
 - [Exclamation Mark](#)
 - [Semi-Colon](#)
 - [Slashes](#)
 - [Sentence Structure](#)
 - [Capitalization](#)
 - [Prerequisites](#)
 - [Post-Requisites](#)
 - [Condition Phrases](#)
 - [Verbiage](#)
 - [Window vs. Page](#)
 - [Tab vs. Section](#)
 - [Introducing Links](#)
 - [Latin Phrases](#)
 - [Abbreviations](#)
- [Headings](#)
- [Lists](#)
 - [Unordered Lists](#)
 - [Bullet](#)
 - [Divider](#)
 - [No Accents/Bullets](#)
 - [Ordered Lists](#)
 - [Markdown](#)
 - [HTML](#)
 - [Numbered List](#)
 - [Alphabetic List](#)
 - [Nested Lists](#)
 - [Description List](#)
- [Images](#)
 - [Standard Images](#)
 - [Inline Images](#)
 - [Markdown](#)
 - [HTML](#)
 - [Image Alt Text](#)
- [Tables](#)
 - [Standard Table](#)
 - [Scroll Overflow](#)
 - [Span Multiple Columns](#)
 - [Adjust Column Widths](#)
 - [Sticky Header](#)
 - [Footnotes](#)
- [Glossary Tooltips](#)
- [Notes](#)
- [Tips](#)
- [Links](#)
 - [Internal Links](#)

- [External Links](#)
- [Links to Vault Help \(Platform\)](#)
- [On-Page \(Anchor\) Links](#)
- [Muted Links](#)
- [Accordion](#)
- [Contacting Services/Support](#)
- [Typographical Conventions](#)
- [How to Reuse Content](#)
- [How to Condition Content for Release Type](#)
- [GitLab](#)
 - [GitLab Command Reference](#)
 - [Fetch Remote Branches](#)
 - [List Existing Branches](#)
 - [Checkout a Branch](#)
 - [Stage All Changes in Current Directory](#)
 - [Commit Changes to Git](#)
 - [Push the Commit to a Branch](#)
 - [Merge Branches](#)
- [Jekyll](#)
 - [Preview Site](#)
- [AWS S3](#)
 - [S3 Buckets](#)
 - [Staging Environment](#)
 - [Production Environment](#)
 - [Okta Authentication](#)
 - [Generate AWS Credentials via Okta](#)
 - [S3 Command Reference](#)
 - [Sync an Entire Folder](#)
 - [Copy a Single File](#)
 - [Remove a Single File](#)
 - [Empty an Entire Bucket](#)
 - [List AWS Buckets](#)
- [Resources](#)

Introduction to the Authoring Environment

On the Vault Safety team, we use Jekyll, a static website generator. To author user docs in Jekyll, we create Markdown/HTML files and edit them using a code editor.

How to Add a New Page

Help pages are stored in the **_docs** folder. Release notes are stored in the **_release-notes** folder. To add a new page, start by locating the folder and sub-folder (if applicable) where the page should go, and create a new .md file.

Naming Files and Folders

Jekyll generates webpage URLs using the filename and file path, so it's important to name files and folders with user-friendly names.

In general, name files with the page title, replacing any spaces with hyphens. For example, a page with the title "About Adverse Event Reports" should be named [about-adverse-event-reports.md](#).

Note: Do not rename existing folders or files without consulting with other writers first. Renaming files will break any existing links to that file.

Add YAML Front Matter

Add YAML front matter to the top of each page to define certain metadata about the page. Start and end the front matter with three dashes.

```
---
title: Page Title Goes Here
subtitle: Write a short description for every page in the subtitle.
layout: doc
tags: [a single tag can go in square brackets]
- separate multiple tags
- using dashes
---
```

Note: You cannot use the Liquid in YAML front matter.

Add a Table of Contents

To add a table of contents to a page, add the following reuse Liquid tag after the YAML front matter and before the page content:

```
{% include sections-in-this-article.md %}
```

Add Page Content

Use the sections in this guide to help you with adding content to the page. You can choose whether to author in HTML or Markdown in most cases, but certain items require one method over the other.

Follow the recommendations in this guide to ensure your content is consistent with the website's stylistic theme.

Update the Navigation Menu

When you add a user help page, you must update the appropriate navigation menu for the page to appear in the site navigation pane. We have a navigation menu for each chapter, located in **`_data/navigation`**.

Because we have multiple applications (Safety, SafetyDocs, Safety.AI), specify which app this new page is a part of in the link.

Locate the menu for the chapter where you want to insert the navigation link, and then add one of the following links:

Add a Top-Level Link

Use the following syntax to add a top-level link:

```
-
  title: {page_title}
  url: {page_url}
  app:
    -{safety | safetydocs}
```

Add a Sublink

Use the following syntax to add a level two (sub) link under a top-level link:

```
-
  title: Top-level link title
  url: {base_url_for_sublink}
  app:
    - {safety | safetydocs}
  subcategories:
    - subtitle: {page_title}
      suburl: {page_url}
      app:
        - {safety | safetydocs}
```

Update the Pagination List (Previous and Next Buttons)

In addition to updating the navigation pane, you must also update the doc pagination YAML map for the document link to correctly appear at the bottom of the previous and next pages. This applies to user help pages only.

The pagination lists are located in the **_data/** folder. There is a list for each application:

- Safety.AI: safetyai-pagination.yml
- SafetyDocs: safetydocs-pagination.yml
- Safety: safety-pagination.yml

Locate where you want to insert the page in the series. The order should match the order of pages in the navigation pane. Note that the structure must be flat; you cannot have sublist items in the pagination list.

Use the following syntax to add a page:

```
-
  title: {page_title}
  url: {page_url}
```

Writing Style

For any topics not covered in this style guide, follow the [Microsoft Style Guide](#).

Titles and Headings

Generally, titles and headings should be specific, but informative enough to guide the reader to specific topics. Consider the following rules when forming a heading:

- Ideally, headings should not exceed 8 words.
- All essential words should be capitalized (e.g. "Add a Product from Your Company Library"). Tools such as [Capitalize My Title](#) are helpful for this purpose.
- Headings should begin with a verb have no gerunds (-ing endings). e.g. "Add a Document".
- There is no need to use colons after headings.
- Construction of headings should be grammatically parallel (e.g. two same-level headings should be "Add a Study Arm" and "Add a Study Arm Product", not "Add a Study Arm" and "[Adding](#) a Study Arm Product"). For more information, visit [Parallel Structure of Headings](#).
- Heading tags should be correct and consistent with the overall page (e.g. h1 for page headings, h2 for section headings, h3 for subsection headings).
- Always include keywords in the heading, because search results are filtered first through headings and then through the general page content; search results prioritize H1 headings, then H2, etc. Functions and main ideas can be added to overviews and introductions. (e.g. "Complete Case Triage" is a better heading than "Determine if a Case requires Expedited Reporting").
- Avoid stacking headings (adding one heading directly below another). Try to add at least one sentence of information after each heading.

Punctuation

The following are some general rules for punctuation. [Grammarly](#) is a great resource for additional information.

Commas

[Commas](#) should be used where the reader would naturally pause in order to register a sentence. However, it is best to remove random commas that interrupt the flow of a complicated sentence.

Remember the following rules about comma usage:

- Always use the [Oxford comma](#).
- Consider using bullet points rather than listing items in a sentence separated by commas.

Colon

A [colon](#) introduces one or more elements in relation to the information preceding the colon. The following are guidelines for using colons on the Safety help site:

- Always precede a colon with a full sentence. *Tip:* An easy way to master this is by making sure the sentence can stand on its own as a complete sentence without the colon.
- Use colons to introduce terms in a list, such as when introducing [field names](#).
 - The first word after a colon in a list should generally be capitalized e.g. Study Number: Enter the sponsor study number.

Exclamation Mark

Avoid using [exclamation marks](#) because they detract from the professional tone of the content.

Semi-Colon

A [semicolon](#) joins two complete sentences without using a conjunction. Semicolons should be used sparingly on the help site since we aim to write in short, uncomplicated sentences.

Slashes

[Slashes](#) are used on the Safety help site to denote "or", e.g. SUSAR/SAE detection.

Sentence Structure

Keep sentences short, clear, and concise.

Sentences should be written in active, not passive voice whenever possible. (e.g. *active*: "You can configure study arms for each study." *passive*: "Study arms can be configured for each study by you .") For more information, visit [Active and Passive Voice](#).

Capitalization

Capitalization should be correct and consistent. On the Safety help site, the following should be capitalized:

- Proper nouns (e.g. Veeva Systems, Vault Safety)
- Names of Vault objects, fields, sections, etc. (e.g. Case, Narrative Document)
- Acronyms and abbreviations (e.g. FDA, E2B, CIOMS)
- User types (i.e. System Admin, Business Admin, Vault Owner)

Prerequisites

In general, prerequisites are introduced in single sentences. e.g. "Before you can create an aggregate report, you must configure the product Reporting Family."

Post-Requisites

In bold, write **After You Finish** on its own line. Then, write the post-requisite steps on a new line. For example:

After You Finish

Configure Reporting Family Members.

Condition Phrases

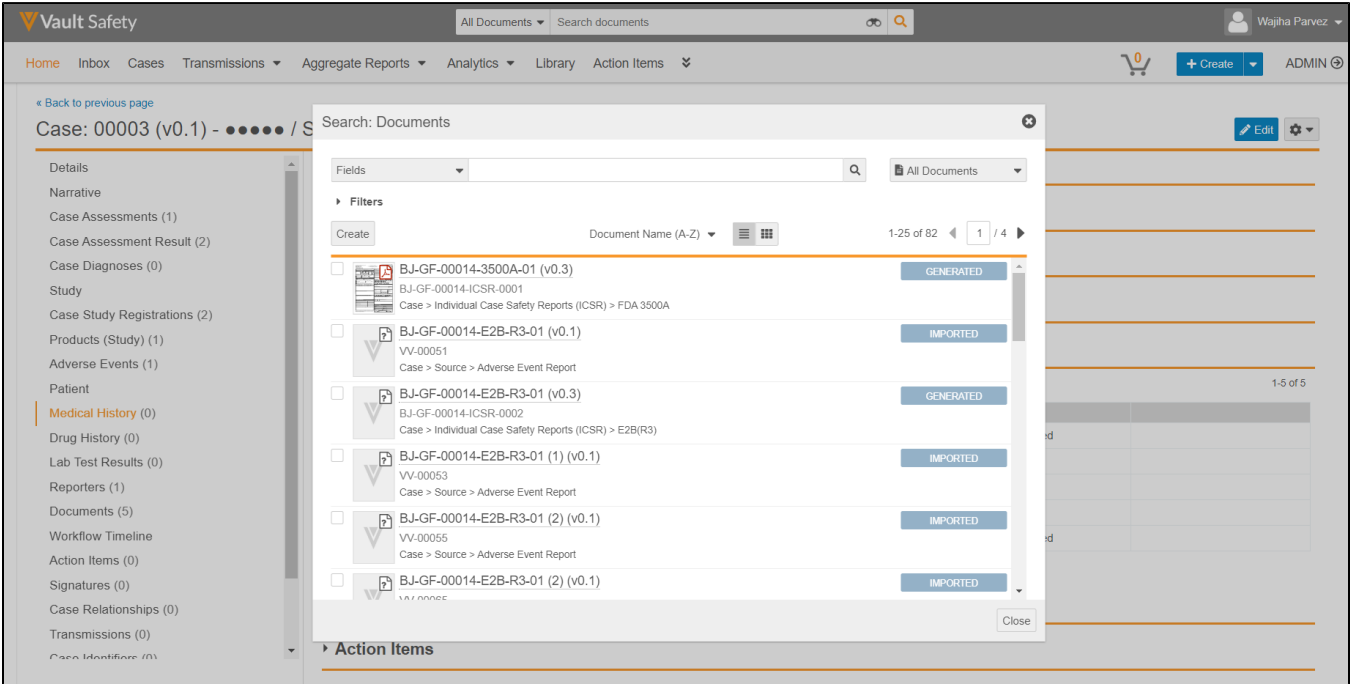
When a statement is only true when a condition is met, introduce the condition first. For example: If an Adverse Event Report is missing information, you can request additional information.

Verbiage

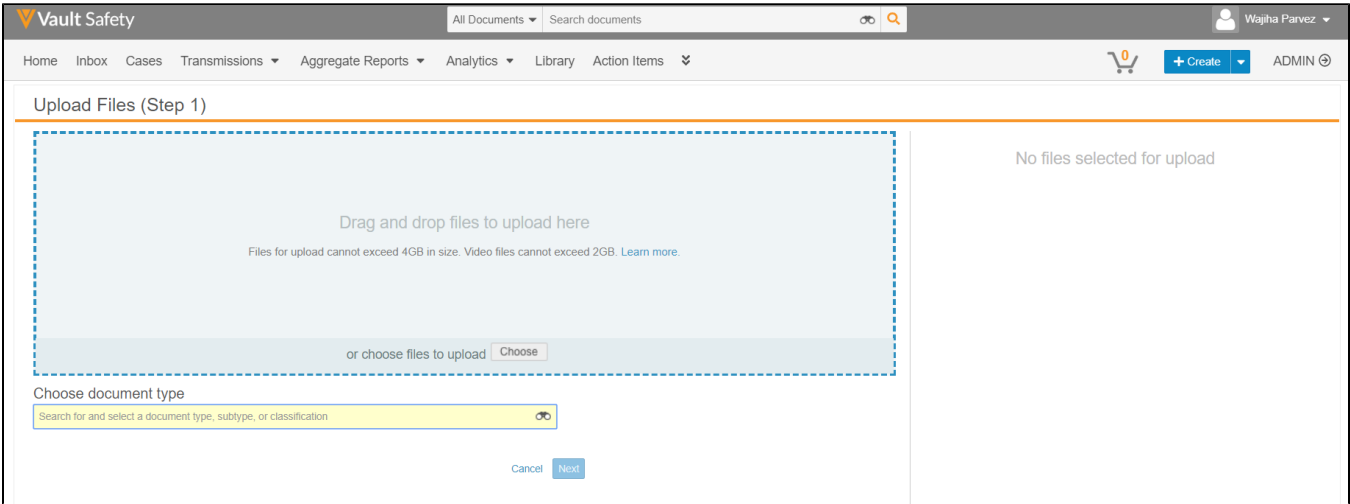
Follow these guidelines to maintain consistency with the rest of the website.

Window vs. Page

If selecting an option opens up a box on the same page, we refer to it as a window. For example, "On the **Search: Documents** *window*, select **Create**."



If selecting an option redirects the user to a new page, we refer to it as a page. For example, "The **Upload Files (Step 1)** *page* appears."



Tab vs. Section

We refer to the options in the menu bar and on any side panes as tabs. For example, "In the **Admin** area, go to the **Users & Groups** *tab*. In the left pane, select the **Vault Users** *tab*."

Business Admin	Logs	Users & Groups	Configuration	Operations	Deployment	Connections	Settings	VAULT
----------------	------	----------------	---------------	------------	------------	-------------	----------	-------

USERS & GROUPS

Vault Users

Active Delegations

Groups

Security Profiles

Permission Sets

Application Roles

Users

+ Create

Active Users

Search Columns

Q

1-11 of 11

Name	Country	Organization	Title	Image	Telephone	Mobile
Application Owner						
Asi Roll						
Emily Rushton						
Eric Woolven						
Florian Letourneux						
James Tan						
Mami Akebono						
Marius Mortensen						
Shanul Srivastava						
Takuto Tsuruoka						
Wajiha Parvez						

We refer to the divisions on a form as sections. For example, "In the **Details** section, complete the following fields..."

Users >

Create User

Domain User

General Info

Address

Details

License Type*

Security Profile*

☐ Domain Admin

Federated ID

Last Login

Security Policy

Salesforce Username

Email Preferences

Application Licensing

Introducing Links

When introducing Learn More links, such as in New Feature descriptions, introduce the "Learn More:" in bold and follow with a bulleted list of links. e.g:

- Learn More:
- Enable Masked Distributions
 - Generate Masked Distributions
 - Manage Narrative Templates

In all other cases, links should be embedded within sentences e.g: "Vault Online Help provides information on general Vault Platform administration".

Latin Phrases

Avoid Latin phrases (verbatim, etcetera, vice versa) as they can be unclear and are not friendly for a global audience. They can also cause translation issues. Find another way of saying what you want to express.

To introduce an example on the help site, say "for example", not "e.g."

Abbreviations

The use of abbreviations should be standard and consistent.

Abbreviations should be spelled out when first mentioned on the page (e.g. "individual case safety report (ICSR)"). The abbreviation can then be used throughout the page.

Headings

You must use Markdown headings for the page's table of contents to pick them up. *Tip:* If you want to intentionally hide headings from the table of contents, use HTML.

In Markdown, create headings using number signs: #. The number of signs indicates the heading level:

Heading Two

Heading Three

Heading Four

Heading one is always reserved for the page title (identified in the YAML front matter). When writing pages, use heading two (##) at the highest level.

If you want to intentionally hide headings from the table of contents, you can write them in HTML. See the [Ulkit documentation](#) for additional styling options.

Lists

When listing three or more items in a sentence, make a list instead. The following are some best practices to consider when making a list:

- Limit lists to nine items or less. Any longer and the list may be overwhelming or easy to get lost in. *Tip:* Break up procedures into multiple, smaller tasks if necessary or make a table instead.
- Every list should have an introductory sentence (e.g. "There are two options available to populate this field:" *followed by the options*).
- The construction of lists should be grammatically parallel. e.g.
 - In the **Comment** field, enter any comments about the decision.
 - In the **Study** field, select the study that monitors the product.
 - In the **Product (Study)** field, select the product.

Unordered Lists

Use a Ulkit list and the HTML element for optimal spacing in unordered lists.

Bullet

Generally, use the uk-list-bullet class for unordered lists:

```
<ul class="uk-list uk-list-bullet">
  <li></li>
  <li></li>
  <li></li>
</ul>
```

Divider

As an alternative to a bulleted list, you can use a divider list, which adds lines between each list item:

```
<ul class="uk-list uk-list-divider">
  <li></li>
  <li></li>
  <li></li>
</ul>
```

This type of list is generally reserved for long or overly wordy lists.

No Accents/Bullets

If you do not want any bullets or dividing accents, simply use the `uk-list` class alone:

```
<ul class="uk-list">
  <li></li>
  <li></li>
  <li></li>
</ul>
```

Ordered Lists

For simple ordered lists without any nesting, you can use Markdown. If you want to nest lightbox images, notes, or sublists, you must use HTML.

Markdown

Start the list by typing the number, followed by a period. Start each number on a new line.

HTML

Use the `` element to add an ordered list in HTML.

Numbered List

The following syntax adds a numbered list:

```
<ol>
  <li></li>
  <li></li>
  <li></li>
</ol>
```

Alphabetic List

The following syntax adds an alphabetic list (useful for substeps):

```
<ol type="a">
  <li></li>
  <li></li>
  <li></li>
</ol>
```

Nested Lists

To nest a list, insert the list within an `` element. Do not nest lists more than two levels deep. The following example shows procedure nested lists:

```

<ol>
  <li>Step One</li>
  <li>Step Two:
    <ol type="a">
      <li>Substep 1</li>
      <li>Substep 2</li>
    </ol>
  </li>
  <li>Step Three:
    <ul class="uk-list uk-list-bullet">
      <li>Bullet 1</li>
      <li>Bullet 2</li>
    </ul>
  </li>
</ol>

```

Description List

Use a description list when listing and describing terms or components. Use the following HTML syntax to add a description list:

```

<dl class="uk-description-list">
  <dt></dt>
  <dd></dd>
</dl>

```

Images

Store images in the /uploads folder. Do not rename images without consulting with the rest of the writing team as it will break any existing references to the image.

Inline images do not use special styling and can use HTML or Markdown syntax. For larger (standard) images, you must use Liquid syntax to get the special styling.

Standard Images

For standard images, use Liquid to leverage the appropriate spacing and shadow styling. Use the following Liquid tag to insert an image from the /uploads folder:

```
{% include image.html img="filename.png" lightbox="true" alt="alt-text" caption="figure caption" %}
```

Note: The *lightbox="true"* text is optional and enables the reader to select the image to display it full-screen. Generally, use lightbox for large images that readers may want a closer look at.

Inline Images

Inline images should be small enough to not be overly distracting or create a noticeable gap in line spacing. As a best practice, for clarity, use images instead of text for user interface elements with no text to identify it. For example, instead of “the **Actions Gear Menu**”, use “the menu”.

Markdown

To add an inline image in Markdown, use the following syntax:

```
![alt-text]({{ '/uploads/image.png' | relative_url }})
```

HTML

To add an inline image in HTML, use the following syntax:

```

```

Image Alt Text

You must include "alt", or alternate, text for every image you add. Alt text should be a brief description of the image, such as "gear icon" or "product page".

Alt text describes an image and is important for a couple of reasons. It is an essential accessibility feature for readers using screen-reading software. Also, if the image ever breaks, the alt text will appear in place of the image.

Tables

Use tables to display information in a way so that the reader can quickly scan and compare various fields or options.

As a best practice, use tables for field reference when you must describe seven or more fields.

Note: Only create tables in HTML. Do not use Markdown.

Standard Table

Use the following syntax to add a basic table:

```
<table class="uk-table uk-table-striped uk-table-responsive">
  <thead>
    <tr>
      <th>Heading</th>
      <th>Heading</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Table Data</td>
      <td>Table Data</td>
    </tr>
    <tr>
      <td>Table Data</td>
      <td>Table Data</td>
    </tr>
  </tbody>
</table>
```

Scroll Overflow

If your table is wider than its container element or would eventually get too big on a specific viewport width, wrap it inside a <div> element with the "uk-overflow-auto" class:

```
<div class="uk-overflow-auto">
  <table class="uk-table uk-table-striped uk-table-responsive">...</table>
</div>
```

Span Multiple Columns

Use colspan on table heading, <th>, or table row, <tr>, elements to define how many columns a cell should span.

For example:

```
<tr>
<td colspan="4">Table Data</td>
</tr>
```

Note: Do not use `rowspan`. Spanning multiple rows messes up the alignment in our tables.

Adjust Column Widths

By default, tables are automatically sized based on how much content is in each column. However, you can enforce proportionate widths on each column.

Set the `"uk-width-x-y"` class on table heading, `<th>`, or table row, `<tr>`, elements to adjust the table column widths.

As a best practice, if there are multiple tables on a page with similar content, size the columns in each table to align with each other.

The [Ulkit documentation](#) provides more information on `uk-width` options and usage.

Sticky Header

For very long tables, you can make the header to stick to the top of the page when the reader scrolls. This isn't officially supported with Ulkit but we have a workaround by placing the header in its own table and removing the margins.

Use the following syntax to add a sticky header:

```
<table uk-sticky="offset: 81" style="background-color: white; z-index: 981" class="uk-visible@s uk-table uk-table-divider uk-margin-remove">
  <tr>
    <th class="uk-width-1-2">Heading</th>
    <th class="uk-width-1-2">Heading</th>
  </tr>
</table>

<table class="uk-table uk-table-striped uk-margin-remove">
  <thead class="uk-hidden@s">
    <tr>
      <th class="uk-width-1-2">Heading</th>
      <th class="uk-width-1-2">Heading</th>
    </tr>
  </thead>
  <tbody>...</tbody>
</table>
```

Note: When using sticky headers, you must set the same width on the header table and the main table to ensure they line up.

Footnotes

Add footnotes within `<tfoot>`. Place `<tfoot>` after the end of `</tbody>` but before the end of `</table>`:

```
<table class="uk-table uk-table-striped uk-margin-remove">
  <thead>
    <tr>
      <th class="uk-width-1-2">Heading</th>
      <th class="uk-width-1-2">Heading</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Body</td>
      <td>Body</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Footer</td>
      <td>Footer</td>
    </tr>
  </tfoot>
</table>
```

Note: To make footnotes span the whole table, add the colspan command inside <td>. Refer to the **Span Multiple Columns** section for more information.

Glossary Tooltips

Use the Glossary Tooltip feature to add definitions to terms that display above the word when hovered over:

Veeva Vault is a true cloud enterprise content management platform and suite of applications specifically built for life sciences.

[source]

le system, Vault Safety

To use the Glossary Tooltip feature:

1. Define terms in the Glossary in the `/_data/glossary.yml` file:

- **Term:** Specify the name of the term. This field is case-sensitive. The term will be used in the liquid tag in the markdown text where the definition will appear.
- **Definition:** Specify the definition that will appear in the tooltip.
- **Url (optional):** Add a link for the definition. The link will appear as [source] in the tooltip.

```
- term: Vault
  definition: Veeva Vault is a true cloud enterprise content management p
  url: https://www.veeva.com/products/enterprise-content-management/
```

Note: Glossary tooltip definitions in the glossary.yml file should be condensed versions of the main Glossary definitions. Tooltip definitions should be no more than 3 sentences.

2. Add the glossary term anywhere in a document with the following syntax:

```
{% glossary [term] %}
```

For example, adding the "Vault" glossary term to in-line markdown text:

```
{% glossary Vault %} Safety
```

Notes

Add notes to call out important information. Use the following syntax to add a note:

```
{% include alert.html style="success" text="This is a note." %}
```

If the note includes a link, use the following syntax, and include the link tag in the note text:

```
<div data-uk-alert>
<p><span class="uk-label" style="background-color:#3895C4;">Note</span><span style="color: #808080"> This is a
note with an <a class="link" href="{{ site.baseurl }}" {% link _references/feature-enablement/masked-
distributions.md %}>Example Link</a>.</span></p>
</div>
```

Tips

Add tips to inform the user of recommended actions.

```
{% include tip.html text="This is a tip." %}
```

If the tip includes a link, use the following syntax, and include the link tag in the tip text:

```
<div data-uk-alert>
<p><span class="uk-label" style="background-color:#f7981d;">Tip</span> This is a tip. </p>
</div>
```

Links

The type of link you should add depends on the location of the page or section you are linking to.

Internal Links

Use the following Liquid link tag along with the document file path (which you can find by tracing the document back to the base site folder) and your relevant link text and Liquid will generate the link:

```
<a href="{{ site.baseurl }}" {% link _docs/process-a-case/coding/code-meddra-terms.md %}>Code MedDRA Terms</a>
```

You may need to link to a heading within a page, such as for Learn More links in release notes. To do so, add that heading to the tag in the format `#this-is-a-heading`. For example:

```
<a href="{{ site.baseurl }}{% link _docs/process-a-case/coding/code-meddra-terms.md %}"#fields-that-support-meddra-coding">Fields that Support MedDRA Coding</a>
```

The benefit of using the Liquid link tag instead of inserting the page URL directly is that Jekyll will fail to generate the site and display an error if there are any broken links. You must include `{{ site.baseurl }}` for Jekyll to prepend the correct site-path, whether `/lr` or `/gr`.

External Links

For external links, add `target="_blank"` to make the page open in a new tab. For example:

```
<a target="_blank" href="https://link.com/">https://link.com/</a>
```

Links to Vault Help (Platform)

To link to Vault Platform Help, use the following syntax:

```
<a href="{{ site.vh-link | append:'/<path>/' }}" target="_blank">Vault Online Help</a>
```

where `'<path>'` is the path to the specific page you want to link to AFTER **wordpress**.

For example, to link to the page <http://vaulthelp.vod309.com/wordpress/admin-user-help/object-lifecycles-admin/configuring-object-workflows/>, the path would be `'admin-user-help/object-lifecycles-admin/configuring-object-workflows/'`, and the link would be:

```
<a href="{{ site.vh-link | append:'/admin-user-help/object-lifecycles-admin/configuring-object-workflows/' }}" target="_blank">Vault Online Help</a>
```

VH-link is a variable set in the site configuration file that points to the appropriate Vault Help site (LR or GR) based on the Vault Safety Help version.

On-Page (Anchor) Links

To link to a place on the same page, add an anchor, and then link to that anchor. For headings, you are not required to add an anchor as they are automatically generated by the Jekyll build process. Replace all spaces with dashes and make all text lowercase for linking to the heading anchor.

1. Create an anchor on the page:

```
<a id="anchor"></a>Here
```

2. Link to the anchor:

```
<a href="#anchor">Go There</a>
```

If you can, add the anchor slightly higher than the section you want to link to. The website navigation bar offsets the anchor location.

Muted Links

Add `class="uk-link-muted"` to make the link less distracting. A muted link appears gray until you hover over it, after which it becomes blue and underlined. For example:

```
<a class="uk-link-muted" href="https://link.com/">https://link.com/</a>
```

Accordion

Use accordions to hide content under a heading with a clickable toggle. This element can be useful when you want to add a lot of supplementary information on a page but not display it all at once.

Use the following syntax to add an accordion:

```
<div>
<div class="uk-container uk-container-small">
<ul data-uk-accordion="multiple: true">
  <li>
    <a class="uk-accordion-title uk-box-shadow-hover-small" href="#">Title</a>
    <div class="article-content uk-accordion-content">Section content</div>
  </li>
</ul>
</div>
```

The data-uk-accordion="multiple: true" attribute allows multiple accordions to stay open at the same time. To force other accordions to close when a different accordion is opened, use data-uk-accordion="multiple: false"

Contacting Services/Support

When adding advice to contact our Veeva Services or Support teams, use the following guidance:

- For services, there is no generic contact. Write "we recommend that you contact Veeva Managed Services to help with . . ."
- For support, we can link them to the [Veeva Product Support Portal](#)

Typographical Conventions

Text Type	Typography	Example
User Interface Element (Window, Page, Tab, Pane, Buttons, Field Names)	Bold Markdown: **Bold** HTML: Bold</Text>	On the Products page, select Edit .
System Message	<i>"system message"</i> For blockquotes, use Markdown: > system message For inline quotes, use HTML: <q>system message</q>	
File Names or Extensions	.FILE Markdown: `*.FILE` HTML: <code>.FILE</code>	
Variables	<i>{emphasized in curly brackets}</i> Markdown: {*emphasized*} HTML: {emphasized}	

Example Input	input	
	Markdown: `input`	
	HTML: <code><code>input</code></code>	

How to Reuse Content

To create referenceable and reusable content, place the content in an HTML or Markdown file in the `_includes/reuse` folder.

To insert the content on any page, you must reference the file using the Liquid include tag:

```
{% include reuse/filename.md %}
```

Note: Do not insert any YAML front matter in the file.

How to Condition Content for Release Type

You can add conditions to make certain content only appear for a type of release, whether general or limited, using the Liquid case tag.

Because the Liquid release-toggle variable is set in the respective LR and GR config files, we can use the case tag for a number of reasons. We will just cover the basics here: how to show and hide content for different release types.

The following example shows how to make different content appear depending on the release type:

```
{% case site.release-toggle %}

{% when 'limited' %}
This content will appear when generating the site using the limited release config file.

{% when 'general' %}
This content will appear when generating the site using the general release config file.

{% endcase %}
```

You can also add sections only for certain releases. For example, if you want to add a known issue with a minor use case to the limited release site but not broadcast it in the general release site. In the following example, the content will appear for limited release builds only:

```
{% case site.release-toggle %}
{% when 'limited' %}
This content will appear for limited release.
{% endcase %}
```

GitLab

GitLab is the content management and versioning system we use for the Vault Safety website files. For information about our branches and merging strategy, see [Git Branch and Merge Strategy \(Online Help\)](#)

GitLab Command Reference

Fetch Remote Branches

```
git fetch --all
```

List Existing Branches

```
git branch
```

Checkout a Branch

```
git checkout { enter the branch name }
```

Stage All Changes in Current Directory

```
git add .
```

Commit Changes to Git

```
git commit -m " { enter a brief commit message to reference the changes you made } "
```

Push the Commit to a Branch

```
git push
```

Merge Branches

<https://www.atlassian.com/git/tutorials/using-branches/git-merge>

Jekyll

Jekyll is a static site generator. We have a script that lets you preview your changes locally before pushing them to the repository.

When you run the following command, Git will prompt you to specify the type of site you want to build. When the build is ready, you can preview your site at <http://localhost>

Preview Site

```
bash serve.sh
```

The preview will automatically reload if any files are changed. Remember to stop the preview with **ctrl+c** otherwise, it will run continuously 😊

Note: All commits to the lr-develop and gr-develop branches are automatically deployed to the respective internal staging site and can be previewed there. See [Git Branch and Merge Strategy](#) for more information.

AWS S3

We host our websites using Amazon Web Services (AWS) S3 and CloudFront. For any help or changes required on the buckets, contact Mike Pereira (mike.pereira@veeva.com).

S3 Buckets

The following table lists the AWS S3 buckets for the Vault Safety online help, for both staging and production.

Staging Environment

The following buckets are for the development and staging environment. To test changes and facilitate SME reviews before pushing the changes to the public site:

Base Site	Limited Release	General Release	Pre-Release
safetyhelp-stage.veeva.com	safetyhelp-stage-lr.veeva.com	safetyhelp-stage-gr.veeva.com	safetyhelp-stage-pr.veeva.com

Production Environment

The following buckets are for the public production site. Only publish updates to this site once they are final and customer-ready:

Base Site	Limited Release	General Release	Pre-Release
safety.veevavault.help	safetyhelp-prod-lr.veeva.com	safetyhelp-prod-gr.veeva.com	

Okta Authentication

Before you can communicate with an S3 bucket, you must generate an Okta authentication certificate.

The certificate expires after a few hours, so you must re-authenticate each day you must perform AWS operations.

Generate AWS Credentials via Okta

Note: These instructions are written for Windows machines.

1. Go to **C:\Users\{user}\.okta**
For example: C:\Users\Emily Rushton\.okta
2. Launch **oktacreds.bat**
The script launches in the command prompt.
3. Follow the prompts to enter your credentials and authenticate with Okta.

Result

If Okta authentication succeeds, the process provides you with temporary credentials to communicate with AWS S3.

S3 Command Reference

This section provides information on the common AWS S3 commands you may need to use.

Sync an Entire Folder

```
aws s3 sync "{FILEPATH}" s3://{BUCKET}/{FILEPATH} --region us-east-2
```

Copy a Single File

```
aws s3 cp {FILE} s3://{BUCKET}/{FILEPATH} --region us-east-2
```

Remove a Single File

```
aws s3 rm s3://{BUCKET}/{FILEPATH} --region us-east-2
```

Empty an Entire Bucket

```
aws s3 rm s3://{BUCKET} --recursive --region us-east-2
```

List AWS Buckets

```
aws s3 ls s3://{BUCKET} --region us-east-2
```

Resources

See the following resources for additional guidance and information:

- [Microsoft Style Guide](#)
- [Veeva Web Style Guide](#)
- [Industry Terminology Glossary](#)
- [UIKit Documentation](#)
- [Docs Jekyll Theme Documentation](#)
- [Capitalize My Title](#)
- [Oh, Shit, Git!](#)
- [Docs to Markdown/HTML Plug-In](#)