Universal Technical Writing Style Guide

Universal Technical Writing Style Guide

1. Purpose and Audience

- This guide defines standards for all product, technical, and user-facing documentation
- Our goals: clarity, consistency, professionalism, and a cohesive brand voice across all content
- Designed to be adaptable for any business or platform

2. Why Consistency Matters

- Reduces cognitive load for readers by maintaining predictable patterns
- Ensures clarity and establishes trust in your brand
- Streamlines the writing and review process
- Creates a professional, unified experience across all documentation

Content Structure and Templates

Documentation Types and Templates

Feature Overview Template

Use this template for introducing new features or products:

```
---
title: About <Feature>
```

```
description: Learn about <short description of the feature/product - max 50 words>.
keywords: ["keyword1", "keyword2", "keyword3"]
---

<Feature/product> is <brief description of what the feature/product is and its purpose - max 50 words>.

<Feature> can handle various levels of complexity:
- <Simple use case 1>
- <Complex use case 2>
- <Conditional use case 3>

## How This Helps
<Provide a general description of benefits - max 100 words>.
Link to additional relevant articles when applicable.

## Feature Architecture/How It Works
The following image illustrates the <feature> architecture:
[Insert architectural diagram with descriptive alt text]

These components are described in the following table:
| # | Component | Description |
|---|-----------|-------------|
| 1 | **<Component 1>** | <Description of component 1> |
| 2 | **<Component 2>** | <Description of component 2> |
| 3 | **<Component 3>** | <Description of component 3> |
| 4 | **<Component 4>** | <Description of component 4> |
```

How-To Guide Template

Use this template for procedural documentation:

```
---
title: "Verb (gerund) + object - example: Configuring Workflows"
description: "[Learn how to accomplish specific task - max 50 words]"
keywords: ["keyword1", "keyword2", "keyword3"]
---
```

```
[Feature Name] enables users to [primary function]. [Feature Name] is particularly valuable for [specific user roles] who need to [specific need or challenge].

## Using [Feature Name]
[Feature Name] provides several key operations that help you manage and optimize your [specific processes]. With [Feature Name], you can:
- [Key capability 1]
- [Key capability 2]
- [Key capability 3]
- [Key capability 4]

## [Primary Process - e.g., Create a Workflow]

### Prerequisites
[List any requirements, accounts, or setup needed]

### **To [accomplish specific task]**

#### For Simple Processes (use numbered lists):
1. [Instruction with imperative verb]
   [Expected result or outcome]
2. [Next instruction]
   [Expected result or outcome]
3. [Continue pattern]
   [Expected result or outcome]

#### For Complex Processes (use step structure):
**Step 1: [Action Title]**
[Detailed instruction and context]
[Insert screenshot if applicable]
*Result: [What happens after this step]*

**Step 2: [Action Title]**
[Detailed instruction and context]
```

[Code block if applicable]

```
*Result: [What happens after this step]*

**Step 3: [Action Title]**
[Detailed instruction and context]

| Icon/Element | Name | Description |
|--------------|------|-------------|
| [icon] | Element name | What it does |
| [icon] | Element name | What it does |

*Result: [Final outcome]*
```

Tutorial Template

For step-by-step learning experiences:

```
## Tutorial: Creating Your First [Process]

In this tutorial, you'll learn how to:
- [Learning objective 1]
- [Learning objective 2]
- [Learning objective 3]

### Prerequisites
Make sure you have:
- [Requirement 1]
- [Requirement 2]

### Part I: [Major Section]
This section shows you how to:
1. [High-level step]
2. [High-level step]

#### **Step 1: [Specific Action]**
From the [location], [specific instruction].
*The [result] loads.*

#### **Step 2: [Next Action]**
```

```
[Specific instruction]
*[Expected outcome]*

### Part II: [Next Major Section]
[Continue pattern for remaining tutorial sections]

### Test Your [Process]
You can test your [process] in multiple ways:
- [Testing method 1]
- [Testing method 2]
```

Structure & Formatting Guidelines

Headings & Titles

- Follow hierarchy: Title Case for all headings (H1 for page titles, H2 for major sections, H3 for subsections)
- Use parallel structure throughout hierarchy; match grammatical forms
- **Examples:**
  - Concept pages: "Understanding Permissions", "Managing API Keys"
  - Task pages: "Create a Report", "Configure Your Workflow"
- For overviews: use gerunds ("Creating Your First Workflow")
- For instructions: use imperatives ("View a Log")

Paragraphs & Lists

- Keep paragraphs concise (2–4 sentences)
- Use bulleted lists for options or non-sequential content
- Use numbered lists for sequential steps or prioritized items
- Ensure parallel structure within lists and capitalize first word
- Use periods only if list items are complete sentences

Step-by-Step Instructions

- Each step starts with imperative verb: "Enter your credentials", "Click Save"
- Align instruction order with UI element order
- Provide prerequisites and expected outcomes when relevant
- Group related steps under subheadings for clarity

Visuals & Media

Screenshots and Images

- All images require descriptive alt text; use PNG for screenshots
- First screenshot shows full screen context, cropped images only for details
- Highlight key elements using clear, high-contrast markings
- Include captions that explain what the image demonstrates

Interactive Elements

- Embed demos and interactive content when available
- Provide fallback descriptions for accessibility
- Include loading states and error handling information

Code Examples

- Use fenced code blocks specifying the language
- Include inline code in `backticks` for UI elements and short snippets
- Provide context for code examples
- Include expected outputs when relevant

Tables & Data

- Use clear, sentence case headers
- Make table columns consistent with standard formatting
- Bold field names and important elements
- Include descriptions for complex data

Language & Style Standards

Voice & Tone

- Friendly, direct, professional, and confident
- Active voice preferred: "The system saves your work" vs. "Work is saved by the system"
- Simple present tense for product behavior
- Future tense only for tutorial outcomes and results

Grammar & Usage

- Use consistent regional standards (specify your preference: US/UK English)
- Spell out numbers zero through nine; numerals for 10+
- Use serial (Oxford) comma for lists of three or more
- Place periods and commas inside quotation marks
- Use colons to introduce lists or explanations

Inclusive Writing

- Gender-neutral language mandatory
- Use "they/them" for unknown individuals
- Avoid assumptions about user knowledge, location, or capabilities
- Write for global audiences when possible

UI References & Links

- **Bold** for UI elements, buttons, and field names
- Use exact UI capitalization
- Navigation sequences: **Settings > Preferences > Advanced**
- Descriptive link text; avoid "click here"
- Include file type/size for downloadable content

Content Enhancement Elements

Callouts and Notifications

Use these consistently across all documentation:

💡 Tip

Helpful shortcuts, best practices, or additional methods

Example: Did you know you can also use keyboard shortcuts for faster navigation?

ℹ️ Note

Important supplementary information for better understanding

Example: This feature requires admin permissions to access.

⚠️ Warning

Critical information about consequences or irreversible actions

Example: Deleting this configuration cannot be undone.

✅ Check (Tutorials Only)

Confirmation of successful completion in learning materials

Example: You should now see the confirmation message appear.

Expandable Sections

Use for detailed examples, optional information, or advanced configurations:

Example Use Case: [Scenario Title]

Expand for detailed walkthrough

- Problem: [What challenge this solves]
- Steps: [Key actions taken]
- Outcome: [Results achieved]

Related Content Sections

Always include at end of articles:

See It in Action

[Embed demos, videos, or interactive tutorials when available]

Related Articles

Group related content by category:

- **Getting Started**: [Foundational topics]
- **Advanced Features**: [Complex implementations]
- **Troubleshooting**: [Common issues and solutions]
- **Best Practices**: [Optimization and recommendations]

Error Messages & User Communication

Error Message Structure

1. **State the problem** clearly in user-friendly language
2. **Explain the cause** if it helps with resolution
3. **Provide solution steps** with specific actions
4. Use present tense, avoid blame, minimize technical jargon

Example Structure:

"We couldn't [action]. [Cause if helpful]. [Clear solution steps]."

Good Example:

"We couldn't save your file. Check that you have permission to write to this location and try again."

Poor Example:

"Error 403: Forbidden access exception thrown during file write operation."

Formatting Standards

Capitalization Rules

- **Headings**: Title Case, bold

- **Table Headers**: Sentence case, bold
- **UI Elements**: Bold, match exact UI capitalization
- **New Terms**: Italicize on first mention, then regular text
- **Filenames**: Exact case as shown in system
- **Placeholder Text**: Lowercase in {brackets} or [brackets]

Numbers, Dates, and Times

- Spell out zero through nine; numerals for 10+
- Use commas for thousands (1,000+)
- Decimals for precision (3.5 inches)
- Consistent date format throughout organization
- 12-hour or 24-hour time format consistently

Quality Assurance

Review Process

- All documentation requires peer review for clarity and consistency
- Test all procedures before publishing
- Verify links and references regularly
- Update screenshots when UI changes occur

Maintenance Guidelines

- Regular review cycles for accuracy
- Version control for all documentation
- User feedback integration process
- Analytics tracking for content effectiveness

Accessibility Standards

- Alt text for all images and media

- Color contrast compliance
- Screen reader compatibility
- Keyboard navigation support

Reference Hierarchy

When style conflicts arise, follow this order of precedence:

1. **This Style Guide** - Primary authority for all writing decisions
2. **Organization Brand Guidelines** - Brand-specific terminology and voice
3. **Industry Style Guides** - Platform or domain-specific conventions
4. **General Style References** - Microsoft Writing Style Guide, Chicago Manual of Style, etc.

This guide should be regularly updated to reflect organizational needs and industry best practices. Encourage team input for continuous improvement.