Docs for Developers

An Engineer's Field Guide to Technical Writing

Jared Bhatti
Sarah Corleissen
Jen Lambourne
David Nunez
Heidi Waterhouse

Foreword by Kelsey Hightower

Apress®

Docs for Developers

An Engineer's Field Guide to Technical Writing

Jared Bhatti
Sarah Corleissen
Jen Lambourne
David Nunez
Heidi Waterhouse

Foreword by Kelsey Hightower

Apress®

Docs for Developers: An Engineer's Field Guide to Technical Writing

Jared Bhatti
Berkeley, CA, USA

Sarah Corleissen
Victoria, BC, Canada

Jen Lambourne
Cornwall, UK

David Nunez
San Francisco, CA, USA

Heidi Waterhouse
Mounds View, MN, USA

ISBN-13 (pbk): 978-1-4842-7216-9
ISBN-13 (electronic): 978-1-4842-7217-6
https://doi.org/10.1007/978-1-4842-7217-6

Copyright © 2021 by Jared Bhatti, Sarah Corleissen, Jen Lambourne, David Nunez, Heidi Waterhouse

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Louise Corrigan
Development Editor: James Markham
Coordinating Editor: Jessica Vakili

Cover designed by eStudioCalamar
Cover image designed by Freepik (www.freepik.com)

Illustrations by Neiko Ng

Diagrams by Tegan Broderick and Eleni Fragkiadaki

Code samples by Eleni Fragkiadaki

Distributed to the book trade worldwide by Springer Science+Business Media New York, 1 New York Plaza, New York, NY 10004. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484272169. For more detailed information, please visit http://www.apress.com/source-code.

Printed on acid-free paper

Praise for *Docs for Developers*

> *"Add documentation" is a step in every product release plan, and "we need more docs" is an action item from every internal developer productivity survey, but it's surprisingly difficult to translate those concise goals into useful documentation. Docs for Developers reveals the repeatable process behind incredible documentation.*
>
> —Will Larson, CTO at Calm, author of *An Elegant Puzzle* and *Staff Engineer*

> *Great documentation is an often overlooked yet critical component for ensuring the success and large scale adoption of a software project. Docs for Developers is a must-read for developers and technical writers who want to rapidly accelerate their ability to create documentation that is easy to consume, brings joy to end users, and is capable of dramatically improving business results.*
>
> —Brad Topol, IBM Distinguished Engineer, Open Technology and Developer Advocacy. Co-author of *Kubernetes in the Enterprise*, and *Hybrid Cloud Apps with OpenShift and Kubernetes*

No matter your starting point, you can find techniques and advice to improve your documentation in Docs for Developers. *This book does for dev docs what The Phoenix Project does for devops - makes your aspirations attainable. The API startup story kept me reading and the cute corgi pictures made me smile.*

—Anne Gentle, Developer Experience Manager at Cisco. Author of the book *Docs Like Code* and website `docslikecode.com`.

Good documentation is a multiplier that helps people onboard and explore software. Docs for Developers guides developers and technical writers to document what their users care about, organize content to help users find what they need, and measure how documentation helps users understand and adopt their software.

—Stephanie Blotner, Technical Writing Manager at Uber

Docs for Devs condenses years of knowledge from multiple industry leaders into a concise, actionable framework. This book guides you from planning to production, with hard won insights on every page. Read it today; your users will thank you.

—Eric Holscher, Co-founder of Write the Docs and Read the Docs

Table of Contents

About the Authors xv

Acknowledgments xvii

Foreword xix

Introduction xxiii

Chapter 1: Understanding your audience 1

Corg.ly: One month to launch 1

The curse of knowledge 3

Creating an initial sketch of your users 4

Defining your users' goals 4

Understanding who your users are 6

Outline your users' needs 7

Validate your user understanding 8

Using existing data sources 9

Collecting new data 10

Condensing user research findings 14

User personas 15

User stories 16

User journey maps 17

Creating a friction log 19

Summary 21

Chapter 2: Planning your documentation 23

Corg.ly: Creating a plan 23

Plans and patterns 24

Content types 25

Code comments 25

READMEs 27

Getting started documentation 29

Conceptual documentation 30

Procedural documentation 31

Reference documentation 35

Planning your documentation 41

Summary 44

Chapter 3: Drafting documentation 45

Corg.ly: First drafts 45

Confronting the blank page (or screen) 45

Setting yourself up for writing success 46

Choosing your writing tools 47

Breaking through the blank page 47

Defining your document’s title and goal 48

Creating your outline 49

Meeting your reader’s expectations 50

Completing your outline 51

Creating your draft 52

Headers 53

Paragraphs 54

Procedures 54

Lists 55

Callouts 56

Writing for skimming 57
State your most important information first....58
Break up large blocks of text....59
Break up long documents....59
Strive for simplicity and clarity....60
Getting unstuck....60
Let go of perfectionism....61
Ask for help 61
Highlight missing content....62
Write out of sequence 62
Change your medium 63
Working from templates 63
Finishing your first draft....65
Summary....66
Chapter 4: Editing documentation 67
Corg.ly: Editing content....67
Editing to meet your user's needs 68
Different approaches to editing 69
Editing for technical accuracy 70
Editing for completeness....71
Editing for structure....72
Editing for clarity and brevity 73
Creating an editing process....75
Reviewing your document first....75
Requesting a peer review....76
Requesting a technical review 77

Receiving and integrating feedback 78
Giving good feedback 79
Summary 81
Chapter 5: Integrating code samples 83
Corg.ly: Showing how it works 83
Using code samples 84
Types of code samples 85
Principles of good code samples 86
Explained 87
Concise 90
Clear 92
Usable (and extensible) 93
Trustworthy 94
Designing code samples 95
Choosing a language 95
Highlighting a range of complexity 95
Presenting your code 96
Tooling for code samples 96
Testing code samples 97
Sandboxing code 98
Autogenerating samples 98
Summary 99
Chapter 6: Adding visual content 101
Corg.ly: Worth a thousand words 101
When words aren't enough 102
Why visual content is hard to create 103
Comprehension 104

Accessibility ... 105
Performance ... 106
Using screenshots ... 106
Common types of diagrams ... 108
Boxes and arrows ... 108
Flowcharts ... 110
Swimlanes ... 111
Drawing diagrams ... 112
Start on paper ... 116
Find a starting point for your reader ... 116
Use labels ... 116
Use colors consistently ... 117
Place the diagram ... 117
Publishing a diagram ... 117
Get help with diagrams ... 117
Creating video content ... 118
Reviewing visual content ... 119
Maintaining visual content ... 120
Summary ... 120
Chapter 7: Publishing documentation ... 121
Corg.ly: Ship it! ... 121
Putting your content out there ... 122
Building a content release process ... 123
Creating a publishing timeline ... 124
Coordinate with code releases ... 126
Finalize and approve publication ... 126

Decide how to deliver content....128

Announce your docs....129

Planning for the future....129

Summary....130

Chapter 8: Gathering and integrating feedback 133

Corg.ly: Initial feedback....133

Listening to your users....134

Creating feedback channels....135

Accept feedback directly through documentation pages....136

Monitor support issues....137

Collect document sentiment....138

Create user surveys....139

Create a user council....140

Converting feedback into action....141

Triaging feedback....141

Following up with users....145

Summary....145

Chapter 9: Measuring documentation quality 147

Corg.ly: Tuesday after the launch....147

Is my documentation any good?....148

Understanding documentation quality....148

Functional quality....149

Structural quality....155

How functional and structural quality relate....158

Creating a strategy for analytics....158

Organizational goals and metrics....159

User goals and metrics....160

Documentation goals and metrics....162

Tips for using document metrics....164

Make a plan....164

Establish a baseline....165

Consider context....165

Use clusters of metrics....166

Mix qualitative and quantitative feedback....166

Summary....166

Chapter 10: Organizing documentation....169

Corg.ly: The next release....169

Organizing documentation for your readers....170

Helping your readers find their way....171

Site navigation and organization....172

Landing pages....176

Navigation cues....178

Organizing your documentation....179

Assess your existing content....179

Outline your new information architecture....181

Migrate to your new information architecture....183

Maintaining your information architecture....184

Summary....184

Chapter 11: Maintaining and deprecating documentation....187

Corg.ly: A few releases later....187

Maintaining up-to-date documentation....188

Planning for maintainability 189

Align documentation with release processes 190

Assign document owners 192

Reward document maintenance 193

Automating documentation maintenance 193

Content freshness checks 194

Link checkers 195

Linters 195

Reference doc generators 196

Removing content from your docset 196

Deprecating documentation 197

Deleting documentation 198

Summary 199

Appendix A: When to hire an expert 201

Meeting a new set of user needs 202

Increasing support deflections 202

Managing large documentation releases 202

Refactoring an information architecture 202

Internationalization and localization 203

Versioning documentation with software 203

Accepting user contributions to documentation 203

Open-sourcing documentation 204

Appendix B: Resources 205

Courses 205

Templates 206

Style guides 207

Automation tools 207

Visual content tools and frameworks....209
Blogs and research....210
Books....211
Communities....212
Bibliography....215
Index....221