LETTERS TO A NEW DEVELOPER

WHAT I WISH I HAD KNOWN WHEN STARTING MY DEVELOPMENT CAREER

Dan Moore

Letters to a New Developer: What I Wish I Had Known When Starting my Development Career

Dan Moore
Boulder, CO, USA

To my ladies:

Pam, Charlotte, and Lucy

Contents

About the Author

Dan Moore has over 20 years of experience as a developer. His roles have included employee, contractor, community member, engineering manager, and CTO. He currently leads developer advocacy at FusionAuth, a Denver company, building software to handle authentication, authorization, and user management for any app. In 2018, Dan started a blog exclusively focused on helping new developers "level up" and has published over 150 posts to help them improve their skills and avoid common mistakes. He resides in Boulder, CO. You can find him on Twitter at @mooreds.

Acknowledgments

Often, I recall a comic from Nathan Pyle. In it a customer is waiting in a shop; the shopkeeper asks "Have you been helped?" And the customer answers "By so many people my entire life."

I, too, have been helped by so many people in my life, my career, and during the writing of this book. My heart overflows with gratitude.

Thank you to all the new developers I've talked with and been educated by. Whether I met you at a meetup, a conference, or online, you've taught me so much.

To my editors at Apress, Louise Corrigan, Nancy Chen, and Jim Markham: thanks for guiding me through this experience.

Thanks to my guest letter writers, who shared their experience, time, and wisdom. I'd especially like to thank those who allowed me to use their letters in this book: Allan Wintersieck, Brooke Kuhlmann, Cara Borenstein, Cierra Nease, Dave Mayer, Jeff Beard, John Obelenus, Josh Thompson, Kim Schlesinger, Mark Sawers, Morgan Whaley, Noel Worden, Rick Manelius, Rishi Malik, Rylan Bowers, and Zach Turner.

The chapter reviewers have my gratitude as well. You challenged my ideas and sharpened my thoughts. I'll lay claim to my mistakes, though. Thanks to Andy Pai, Brian Timoney, Corey Snipes, Emerson Loustau, Ian Sprod, Kendall Miller, Kim Schlesinger, Marty Haught, Mike Gehard, Miles Matthias, Ned McClain, Pam Moore, Rylan Bowers, and Steve Ham.

So many of the skills required to be a developer are learned on the job. I've been taught so much by my colleagues, managers, and reports. I'd like to acknowledge four of my bosses in particular.

Bryan Buus, who took a call from a college senior during frenetic preparations for the Kentucky Derby. Thank you for introducing that new developer to the XOR family.

By watching Anthony Francavilla and Lane Horning, my eyes were opened to the power, strength, and fragility of business partnerships. I also learned about the connection between deep business domain knowledge and great software. Lane also gave me my first swing at engineering management.

Acknowledgments

My thanks to Ashley Colpaart, who shared her vision with me and took me along on an adventure. Together we survived the crucible of starting a boot-strapped software business.

I also want to thank my family for encouraging and supporting me throughout my career. Special shout-out to my parents for forcing me to work in their business as a middle schooler, where I first saw the power of software to alleviate human toil.

Thank you to Charlotte and Lucy for giving up your dad on weekends and weeknights as I worked on this book. I'm thrilled to be your father.

Finally, thank you to my wife, Pam. She's my biggest supporter. She gave me the time to write this and pushed me to start when I hesitated. I love you.

Introduction

This book is full of lessons I've learned. I wrote it to help new developers like you. When I started developing software, I had many assumptions. Over the last two decades of working in companies both big and small, alongside tens of thousands of employees and as a startup cofounder, I saw how my assumptions were incorrect.

During that time, I also was introduced to specialties and disciplines that I never imagined were part of software development, and my eyes were opened further.

It was not always easy to learn. I made missteps. I was humbled by teammates, managers, and clients. More than once, I had to sit down at the computer and grind.

I wrote this book to share all this with you.

However, this is not a book about a certain technology or a specialized program you might use. Sure, there is an entire chapter about tools a developer should eventually master, but specific technologies are beside the point. If you are looking for a book to help you understand the latest JavaScript framework, microservices architecture, or DevOps platform (or, for the ambitious, all three!), please buy a different book.

This book is, instead, about principles and practices that never go out of style. These were germane two decades ago and will be relevant far into the future. A lot of what I'll cover are called "soft skills," but they're pretty hard to learn. As far as I know, no college or bootcamp offers "How to Make Mistakes 101." It was easier for me to learn how to program a computer than how to collaborate with a team. Computers get faster every year, but people change every day. And they matter. Software exists only because humans shape it, need it, and pay for it.

Among other topics, this book includes practices to help you level up as a developer, tools to learn, the right way to ask questions, and the role of community in software development.

Over many years, I have learned what it takes to be a professional software developer. While I think anybody working in software can benefit from this book, I wrote it for three audiences: new developers, people considering software development as a career, and mentors.

For new developers

You are new to the software development job market. Perhaps you have completed a bootcamp or college degree. You may refer to yourself as an entry-level or junior engineer.

While everyone's background and skills growth happens at different speeds, new developers generally have less than five years of professional experience. Many new developers are worried about their abilities, don't feel welcome, and have a difficult time finding that first job.

But as an industry, we *need* more new developers. There are so many problems with which software can help. Companies want experienced engineers, but all the senior developers I know started out as new developers. A senior engineer is just a new engineer seasoned with gaffes, education, and time.

For new developers, this book will help you avoid missteps I've made. It also introduces you to disciplines beyond coding critical to success. While programming is crucial for any software product or service, there is much more required to deliver an application.

For anyone considering software development

If you're not sure if software engineering is right for you, this book offers perspectives on how to succeed.

I've intentionally kept the barriers to the layperson low with limited technical jargon. Only a few technologies are discussed, and those sections can be skipped. If you are thinking about becoming a developer, I'd recommend buying this book *and* a book about programming.

Giving a computer commands that it can execute is an important skill for any software developer. But software engineering is so much more. You must know what to build, how to work with your team, and how to maintain your systems.

For mentors

If you are mentoring a new developer, this book can serve as a discussion guide. Because each chapter has letters approaching a theme from different angles, you and your mentee will find it useful for focused mentoring sessions.

As an experienced developer, you'll of course bring your own insights and experience to each topic, from your debugging process to the value of an online community for continuous learning.

And, of course, you may have had a different experience than what I share. Such contrasts are a jumping-off point to discuss the diversity available in a software development career.

Format

This book is a conversation.

Each chapter consists of letters around a theme, written from me to you. This format allows me to approach a topic from different perspectives. At the same time, each letter is easy to read in a single sitting. I read before bed and wanted to write a software book that would neither keep you up nor put you to sleep.

To give the reader a broader view, I have invited other engineers and professionals, from recent bootcamp graduates to CTOs, to contribute their viewpoints. You'll see their letters interspersed among mine, signed by the author.

Thank you

Thank you very much for giving me your attention and time. The lessons from this book will accelerate your software development career. You will learn how to make decisions, timeless tools, and who should manage your career (spoiler alert, it's on you).

In reading this book, you'll learn what I did wrong. And what I did right. You'll make mistakes—trust me, I know! But you will make fewer errors than I did.

Finally, *welcome*!

Welcome to the software development world. You belong here. Just like the real world, it's delightful and dizzying, frustrating and fantastic, grungy and gorgeous.

Welcome, new developer. Welcome.